Question Answering For Machine Reading Alzheimer's Task: Team 4

Rajarshi Das[1], Chenhao He[1], Chenying Hou[1], Alen Lukic[1], Chen Wang[1]
[1]Carnegie Mellon University, Pittsburgh, PA 15213
{rajarshd, chengah, chenyinh, alukic, chenw1}@andrew.cmu.edu

Abstract

These are the working notes describing the process of improving an existing question-answering pipeline for the CLEF 2013 Question Answering for Machine Reading Alzheimer's Task. By performing error analyses on the baseline system, we were able to incorporate improvements into the query formulation, retrieval, and scoring phase of the pipeline which resulted in a 15% relative performance increase on the blind test set.

1 Introduction

This report summarizes the task of analyzing a question-answering system [1] built for the CLEF 2013 Question Answering For Machine Reading (QA4MRE) Alzheimer's Task [2]. This system reads a document which addresses Alzheimer's disease and then attempts to answer ten multiple choices questions about the document. Features from the question, the answers, the document itself, and the background Alzheimer's Disease Literature Corpus[1] (ADLC) are extracted and used to score and rank the answer choices.

Section 2 provides a brief overview of the system and discusses its baseline performance. Section 3 discusses the error analysis we performed on the baseline system to determine where errors might occur. Based on this analysis, we chose to focus on query expansion, term weighting, and retrieval expansion as potential sources of end-to-end system performance. Finally, section 4 discusses the evaluation metric and the performance of our final pipeline.

2 Baseline System

Figure 1 outlines the architecture of the baseline system. The pipeline is built on top of the Apache UIMA framework.[2] A document reader reads the documents from disk and prepares them for processing. A series of annotators create a Solr[3] annotation index for each document and its respective question and answer set. A query string is formulated for each question and is used to retrieve sentences from the document using the Solr index. Each candidate answer is then scored using features extracted from the quetion, the answer itself, and the retrieved sentences.

The Alzheimer's Task uses the c@1 evaluation metric to measure system performance, which is defined as

$$c@1 = \frac{c + u(\frac{c}{n})}{n}$$

where c is the number of correctly answer questions, u is the number of unanswered questions, and n is the number of total questions. The performance of the baseline system is reported as $c@1 = 0.4$.

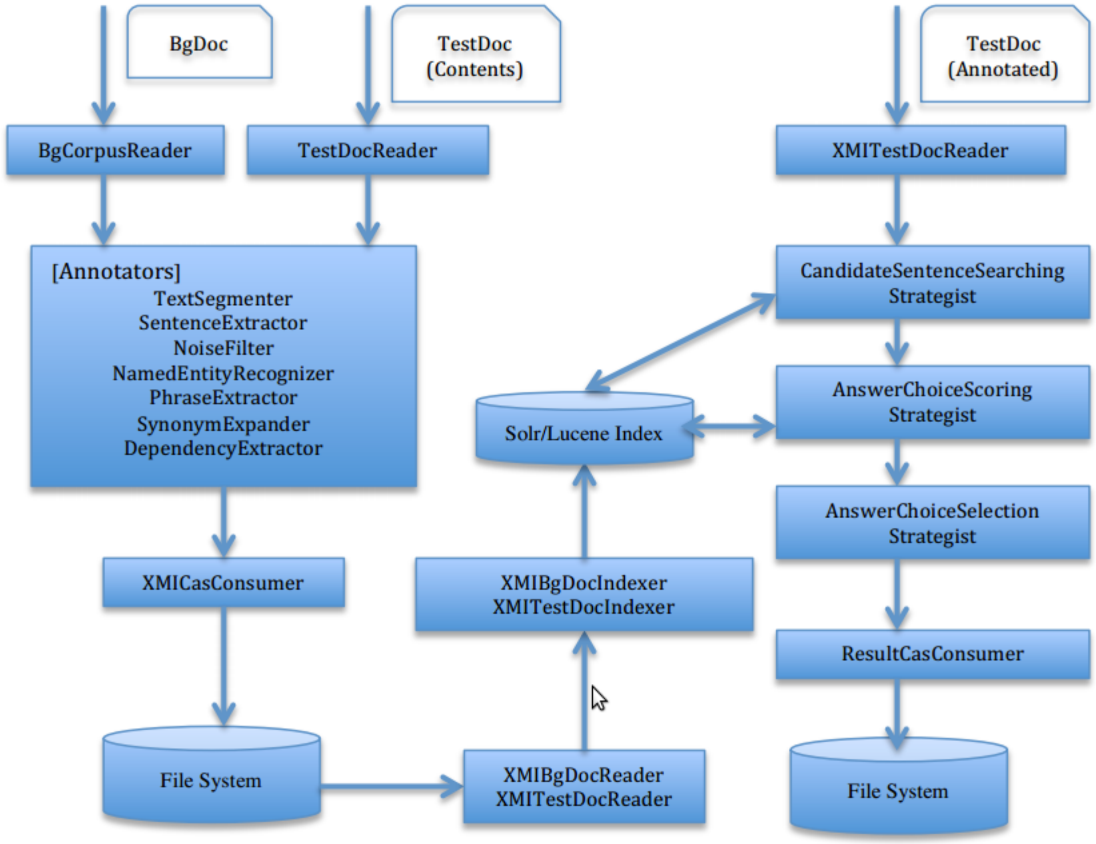

Figure 1: Baseline system architecture [1]

3 Error Analysis & Improvements

We began our error analysis by running the provided baseline system without any modification. An initial run of the pipeline yielded a performance of $c@1 = 0.175$. Closer inspection revealed that some of the annotators in the system specification were absent from the provided code; most notably, the AnswerChoiceCandAnsPMIScorer annotator. Reincorporating this annotator into the pipeline increased the performance of the system to $c@1 = 0.275$.

We then analyzed each phase of the pipeline in more detail through the use of logging tools which enabled us to see the actions taken by the system explicitly. This analysis yielded the following observations:

- All noun phrases and named entities were weighed equally during the candidate answer/retrieved sentence similarity scoring phase.
- Answer-bearing sentences in the document were often in close proximity to the sentences retrieved by Solr, but were not actually retrieved themselves.
- Query strings were formulated using terms from the question only.

Our attempt to address each of these issues is outlined below.

[1] http://celct.fbk.eu/ResPubliQA/index.php?page=Pages/bg_collection_pilot.php
[2] http://uima.apache.org/
[3] http://lucene.apache.org/solr/

3.1 Corpus Expansion and Term Weighting

Clearly, some terms in the query string are more important than others. Our intuition regarding the ADLC corpus was that many of the important terms in this corpus would not appear commonly (or at all) in a corpus which approximates a distribution of vernacular English. We chose the 20newsgroups[4] corpus as an approximation of such a distribution.

We constructed a global frequency table t_g for the 20newsgroups corpus and a local frequency table t_l for each of the Alzheimer's task documents. We assigned a weight w_t to each query term t using the following formula:

$$w_t(t) = \begin{cases} \log(t_l(t)) - \log(t_g(t)) & \text{if } t_g(t) \geq 1 \\ \log(t_l(t)) + 10 & \text{otherwise} \end{cases}$$

This formulation heavily favors terms which appear locally but not globally, in harmony with the intuition that important terms in the biomedical domain are unlikely to appear in a corpus of vernacular English terms. The incorporation of this improvement alone in the pipeline increased the performance of the system to $c@1 = 0.325$.

3.2 Retrieval Window Expansion

Often, Solr retrieves sentences near answer-bearing sentences, but does not retrieve the latter sentences themselves. This is likely due to the lack of coreference resolution in the system. Rather than focusing a considerable amount of effort into a coreference resolution algorithm, we chose to take a simplified heuristic approach: we expanded the set of sentences retrieved from the index by including sentences within a 1-sentence ”window” of each initially retrieved sentence; this expanded the size of the retrieved sentence set by a factor of 3, and the incorporation of this improvement alone in the pipeline increased system performance to $c@1 = 0.325$.

3.3 Query Expansion

Initially, the Solr query strings were formulated using only terms from the question. However, this is obviously a suboptimal formulation, as the answer set may contain many relevant terms not found in the question. We adopted a straightforward query expansion approach and augmented the query with terms from the answer set. This augmentation alone improved system performance to $c@1 = 0.325$.

4 Evaluation & Results

We tested our pipeline on two datasets: the QA4MRE 2012 Alzheimer's Task documents with a gold standard labeled answer set, and the QA4MRE 2013 Alzheimer's Task documents, for which we found the gold standard labels. We evaluate our system using the $c@1$ metric. Since our system answers all questions, the $c@1$ metric is equivalent to an accuracy measurement.

QA4MRE: 2012 Data (Baseline)

	Doc. 1	**Doc. 2**	**Doc. 3**	**Doc. 4**	**Mean**
$c@1$	0.2	0.1	0.1	0.3	0.175

QA4MRE: 2012 Data (Improved)

	Doc. 1	**Doc. 2**	**Doc. 3**	**Doc. 4**	**Mean**
$c@1$	0.4	0.5	0.4	0.2	0.375

QA4MRE: 2013 Data (Improved)

	Doc. 1	**Doc. 2**	**Doc. 3**	**Doc. 4**	**Doc. 5**	**Doc. 6**	**Doc. 7**	**Doc. 8**	**Mean**
$c@1$	0.67	0.1	0.47	0.4	0.4	0.2	0.53	0.53	0.46

[4] `http://qwone.com/~jason/20Newsgroups/`

5 Conclusions

Via relatively straightforward improvements to the pipeline, we achieved a 15% relative accuracy improvement over that reported by the baseline system. Achieving this increase without needing to fundamentally alter the architecture or type system of the pipeline is a tribute to the versatility and extensibility of UIMA. The incorporation of more sophisticated annotation and retrieval techniques could potentially increase system performance even more, and merits future consideration.

References

[1] Patel, A., Yang, Z., Nyberg, E. and Mitamura, T. Building Optimal Question Answering System Automatically using Configuration Space Exploration (CSE) for QA4MRE 2013 Tasks. *In Proceedings of CLEF 2013, 2013.*

[2] Morante, R., Krallinger, M., Valencia, A., and Daelemans, W. Machine Reading of Biomedical Texts about Alzheimer's Disease. *In Proceedings of CLEF 2013, 2013.*