Josue Balandrano Coronel

A Comparative Approach to Question Answering Systems

In this paper I will analyze three different algorithms and approaches to implement Question Answering Systems (QA-Systems). I will analyze the efficiency, strengths and weaknesses of multiple algorithms by explaining them in detail and implementing them with various contexts. With this I will present numerous ideas that can be used to create a truly open context QA-System. The various algorithms and approaches will be focused on complex questions. Complex questions are usually verbose and the context of the question is equally important to answer as is the question itself. Such questions represent an interesting problem in the field because they can be answered and written in a number of distinct ways. Also, the answer structure is not always the same and the QA System needs to compensate for this. The analysis of complex questions differ between contexts. Contexts are the topic to which a complex question belongs, e.g. Biology, literature, etc… The analysis of the answer also differs according to the corpus used. Corpus is a set of documents, belonging to a specific context, where we can find the answer to a specified question. I will start by analyzing in depth various algorithms and approaches. I will then use multiple contexts and corpus to analyze the performance of different algorithms. And to finalize, I will present various ideas to improve the previously mentioned approaches.

Chapter I

Motivation

In this paper I will analyze three different approaches to answer extraction in QA-Systems for complex questions. I will analyze the approaches as well as implement them using different corpus. I will then propose improvements on this approaches, hopefully making one more step into a truly open context QA-System.

The motivation for this paper is to better understand QA-Systems to eventually approach a good implementation for an open context system. The QA-System subject brings together multiple AI topics. The first step of a QA-System is to analyze a natural language input, this is called Natural Language Processing. The second step of a QA-System is to create a corpus in which it will search for the answer. There are different approaches for this, we can explore multiple machine learning algorithms for this. Finally the QA-System has to retrieve multiple passages and then construct an adequate answer for the inputted natural language question. This last step uses a mix of NLP, machine learning algorithms and statistical analysis.

As we can see QA-Systems constitute an angular stone in Artificial Intelligence. This can be seen as one way to communicate with artificial entities. There has been some progress in this topic, one of the best examples of a fully functional QA-System is Watson from IBM. This system is capable of answering factoid questions in an average of three seconds. The IBM team is currently trying to apply Watson to different, more complex, questions and corpus. As we move to complex questions we move to a more real, and intuitive, communication. The type of questions we, as human beings, use to interact with each other are usually complex questions. The main difficulty when processing this type of questions is to fully understand the context of the possible answer as well as the meaning of the question itself. Even when asking questions to another human being we tend to make follow-up questions to fully understand what the inquirer is referring to.

It is evident that by being able to create a truly open context QA-System we are making a step closer to developing a truly Artificial Intelligent Entity. In this paper I will propose various improvements in order to, in the future, arrive to an open context QA-System.

Background

Since the early 1960 there has been research in Natural Language Processing. This research had a tremendous boost in the mid 70’s when the U.S. Government invested a lot of resources in NLP research. Starting in 1992 the Text Retrieval Conference (TREC) co-sponsored by the National Institute of Standards and Technology (NIST) has been a great platform for some of the best improvements in NLP and general information retrieval algorithms and approaches. One of the best examples of fully implemented QA-Systems is Watson by the IBM group. IBM has offered the technology behind Watson as a platform for anyone that would like to use it to apply it to different problems. Watson was first developed to compete with a Jeopardy champion, in the last few years the IBM team have been trying to implement Watson to help doctors diagnose patients. Diagnosing patients is a very important problem which has been studied for several years now. Currently there are a few implementations of QA-Systems directed to helping doctors diagnose patients (askHermes, UpToDate).

The basic structure of a QA-System constitutes of three main blocks. The implementation of this blocks can vary tremendously. The first block is the NLP block, where the system process and analyze the input question, which is in natural language form. When working with complex questions there are different approaches for the processing of the question. The main problem here is to convert from syntax to meaning. Some approaches have even explore the follow-up questioning route. Other approaches try to classificate the question in subtopic so the answer extraction will be within a smaller corpus. The second block is the document retrieval part of the system. Here the QA-System will retrieve a set of document where the possible answer might exists. This has been done in different manners. The most simple of them is to retrieve any document which contains any of the keywords subtracted by the question inputted. This is usually done when working with simple, straight, questions. When dealing with complex questions this becomes more difficult since a complex questions is, more often than not, verbose and we could, possibly, subtract a large set of keywords from it. Another approach is to create regular expressions which will represent the type of answer we are searching for, the type of answer will be set by the type of question the system gets as input. The more complex and effective methods are the ones leveraging on machine learning algorithms.

Clustering, Tree Searches and Bayesian Classifications are some of the algorithms used to retrieve a good set of documents to retrieve the answer from. The last block handles the passage retrieval and answer output. This can be seen as two separate blocks but it depends on the complexity of the processes. Usually passage retrieval resembles the summarization of documents. There are also different machines algorithms used to retrieve the correct passages. This is done because when answering complex questions the answer usually doesn’t lie in one single document. The various machine learning algorithms are used to identificate the correct context and meaning of the passages. Once the correct passages are retrieved the system constructs the answer.

I have presented a couple of QA-System focused on complex questions, but they are also, primarily, focused on medical questions. Complex questions come in different contexts. Temporal questions are particularly interesting since the answer is usually an explanation of a timeline. We can see that the answer will lie in multiple different documents and the summarization of this is not simple at all. There’s also instructional questions, where the answer is usually a series of steps. A QA-System for this might be able to retrieve the answer in just one document but it is always wise to compare the answer with multiple different documents to weight its veracity. This is another interesting challenge in QA-Systems.