Published in Working Notes for CLEF 2013 Conference, Valencia, Spain, September 23-26, 2013. CEUR Workshop Proceedings, vol. 1179 which should be cited to refer to this work.

Using a Question-Answering Approach In Machine Reading Task Of Biomedical Texts About The Alzheimer Disease

Dina Vishnyakova[1], Julien Gobeill[2] and Patrick Ruch[2]

[1] University and University Hospitals of Geneva, Division of Medical Information Sciences, Geneva, Switzerland

dina.vishnyakova@unige.ch

[2] HES-SO/University of Applied Science Geneva, Information Science Department, Geneva, Switzerland

{julien.gobeill, Patrick.ruch}@hesge.ch

Abstract. For the machine-reading task of biomedical texts about the Alzheimer disease we have used a Question-Answering approach by adapting functionalities of Question-Answering (Q-A) engine EAGLi. We didn't involve any other Natural Language Processing method. As a knowledge store we used the biggest resource of biomedical literature - MEDLINE. Our final results showed that the best run was without using the filter of "stop words" in queries. Run 1 and Run 2 provided answers to all 40 Question, while Run 3 and 4 provided answers to 5 questions; Run 5 answered to 6 questions. These results can be tentatively explained by the limits of the Boolean search we chose in the Q-A engine.

1 Introduction

We report on the original integration of an automatic question-answering pipeline that we have developed to perform machine-reading task of biomedical texts about the Alzheimer disease. This task can be basically described as an «open book test» where questions are designed to teach how to use a provided ability in order to find the required information. Regularly, the questions on an open book test are about explanation, evaluation, or comparison of things from the text. The answer to such questions

will not appear in a single paragraph in your text-or even on a single page. The test, when performed by the user or by an automatic reading system can be simplified into the following workflow:

1. Retrieval of documents given a particular query (a particular question) in a particular document repository;
2. Selection of articles: selection of a subset of articles is based on the title and the abstract;
3. Reading of a particular article (or Passage retrieval for an automat). This step is essential for navigating through the information and for the detection of relevant patterns.
4. Extraction of information: a particular passage is analyzed to obtain a representation of the level of entities such as proteins, diseases, methods used to generate a particular result (e.g. yeast 2-hybrid), evidence codes (e.g. automatic inference, direct interactions…);
5. Feedback: this step is optional; it aims at using the generated annotation to improve or refine the search initiated in step 1.

The Organizers of the test performed step 1 and Step 2 by providing articles and questions. Step 3 and Step 4 form the core of the current test where one should know the basic answer to the question and look for the information from the defined resource that will support the answer. Step 5 is often ignored by designers of text mining systems and was not mandatory for the reading test.

The system we designed tentatively covers all these steps. In order to pass the test we have used a Question-Answering approach. As a search tool we have used a Question-Answering (Q-A) engine called EAGLi [1]. It is a question-answering and search engine with terminology-powered navigation and knowledge extraction skills (Gene Ontology, Swiss-Prot keywords, Protein-protein interactions, Medical Subject Headings...) [2][3][4], see Figure 1. The role of library is played by the biggest resource of biomedical literature - MEDLINE. This resource contains journal citations and abstracts for biomedical literature from around the world.

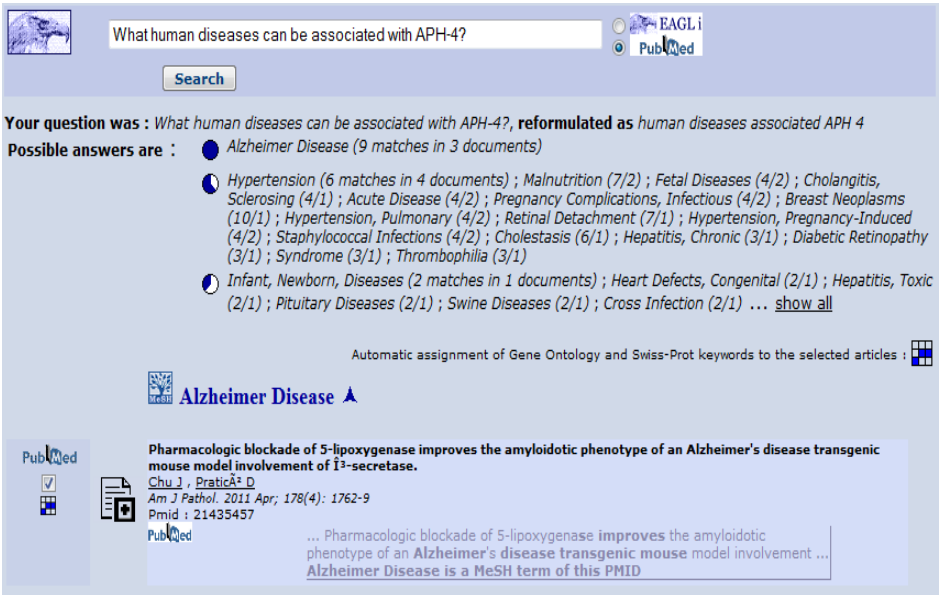

Fig 1. Example of EAGLi interface. Here, the user can provide a question to EAGLi and as a response receives a list of answers. This list is ranked by number of answer matches in documents.

2 Data and Methods

A background collection of the Alzheimer's Disease Literature Corpus, and test documents about Alzheimer's disease were provided. The test set included 4 reading tests (four full articles about Alzheimer disease). Each reading test had one single document, with 10 questions and a set of five choices per question. In total there were 40 questions and 200 choices/options.

We didn't use directly any method of Natural Language Processing area. We have used Boolean method [4] [5] of querying EAGLi [1]. Our approach use the possibility of EAGLi engine to retrieve relevant citations in a given collection from the input query. As an input for EAGLi Q-A engine we generated queries constructed by a conjunction of given question and provided possible answers to this question (if a question has 5 answers it means that as input to EAGLi our system will construct 5 queries of "Question+AnswerX" type, where X is an id of a provided answer). The performance optimization of EAGLi supports utilization of "stop words" function [5] in order to refine the list of returned documents. On the output we receive the number of documents returned regarding the provided query. The system votes for the answer, which returns most documents from MEDLINE. The general workflow of the system is shown on the Fig.2.

We have generated 5 runs in total. Each run represents different combination of the parameters, See Table 1.

Table 1 Generated runs. Here, DIFF is the difference between maximum number of documents returned by one of the queries and average number of documents returned by other queries regarding the same question.

Run	Stop Words filter is applied?	DIFF
1	YES	Not applied
2	NO	Not applied
3	YES	<5%
4	YES	<2%
5	NO	<5%

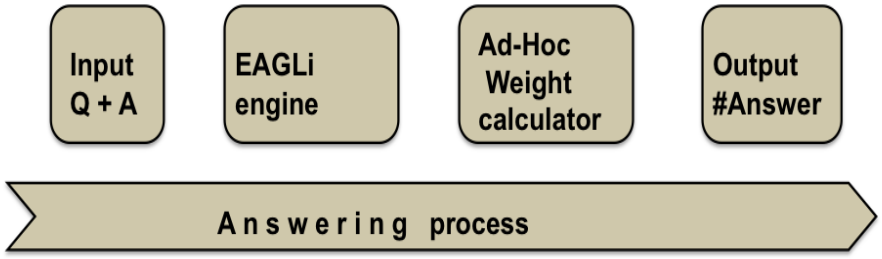

Fig. 2. The workflow of the Q-A approach. Here, *Input Q+A* is a generator of queries based on the conjunction of question with provided answers. Afterward generated queries one by one are passed to *EAGLi engine*. Provided responses from *EAGLi engine* are passed to *Ad-Hoc Weight Calculator* to compute the final answer and finally *Output* delivers an identification of the computed answer.

3 Results and Conclusion

In total we have submitted 5 runs, see Table 2. Run 1 was using "stop words" filter and answered 4 questions correctly out of 40; Run 2 was without "stop words" filter and answered 5 questions correctly; Run 3 was the same as Run 1, but our sys-

tem provided no answers if DIFF was less than 5%. This run provided 0 correct answers. Run 4 is the same, as Run 3, but DIFF was less than 2%. This run returned as well 0 correct answers. Run 5 was without "stop words" and provided no answers if DIFF was less than 5%, but surprisingly it provided 1 correct answer.

Table 2. Results of the submitted runs.

Run	Number of Answered Questions	Number of Correct answers
1	40	4
2	40	5
3	5	0
4	5	0
5	6	1

Our results showed that the way we have designed our system did not support the process of text understanding. The current combination of Q-A functionalities was not able to answer correct to all questions. Although current results seem suggesting that QA can help in machine reading tasks by providing access to relevant contents, it is worth noticing that the effectiveness of our system could be improved by specializing some of the components and most probably by changing Boolean search mode to vectorial search mode. When designing the system, we somehow customize a rather generic text-processing pipeline to answer the specific needs of QA4MRE task.

Considering that some of questions had no answer it implies that Boolean mode was an impropriate choice. Indeed, the Boolean mode does not process negational information but provides documents with given answers. At the same time, the utilization of the vectorial mode [6] could retrieve documents by their similarity to the provided one.

We still believe that it is possible to achieve good results in the current task of QA4MRE by supporting results with Q-A engine usage. Finally, we plan to further investigate how a question-answering engine can be integrated into a machine reading tasks, in particular to address situations when the provided article has no explicit answers.

References

1. Gobeill, J., Tbahriti, I., Ehrler, F., Ruch, P. "Vocabulary-Driven Passage Retrieval for Question-Answering in Genomics." *TREC 2007*, 2007.
2. P., Ruch. "Automatic assignment of biomedical categories: toward a generic approach." *Bioinformatics*, no. 22 658-664, 2006
3. Ruch P., Boyer C., Chichester Ch., Tbahriti I., Geissbühler A., Fabry P., Gobeill J., Pillet V., Rebholz-Schuhmann D., Lovis C., Veuthey A-L. "Using argumentation to extract key sentences from biomedical abstracts." *Medical Informatics* , no. 76 : 195-200, 2007
4. Salton, Gerard, and Harry Wu Edward A. Fox. "Extended Boolean information retrieval." *Communications of the ACM*, 26, no. 11 ,1983.

5. D. Manning, Prabhakar Raghavan and Hinrich Schütze. "Introduction to Information Retrieval,." *Cambridge University Press.*, 2008.
6. Beyer, Kevin, et al.. "When is “nearest neighbor” meaningful? " *Database Theory—ICDT’99:* 217-235, 1999