A framework of evaluation for question-answering systems

Sarra El Ayari *, Brigitte Grau * **

LIMSI - CNRS *
ENSIIE **
{sarra.elayari,brigitte.grau}@limsi.fr

Abstract. *Evaluating complex systems is a complex task. Evaluation campaigns are organized each year to test different systems on global results, but they do not evaluate the relevance of the criteria used. Our purpose consist in modifying the intermediate results created by the components and inserting the new results into the process, without modifying the components. We will describe our framework of glass-box evaluation.*

Key words: glass-box evaluation, question-answering system, relational database, framework, relevance of criteria

1 Introduction

Evaluating a complex systems like question-answering systems is a complex task. Some studies about spoken language dialog systems present methodologies for evaluating systems composed with different modules, where no common accepted architecture exists [2]. It is the same problem with question-answering systems, where the architecture depends on the strategy used. Consequently the difficulty of having an evaluation of the strategies used is evident.
In order to evaluate question-answering systems, evaluation campaigns are organized each year to test different systems on a same task (TREC[1], CLEF[2], NTCIR[3]). Systems have to extract precise answers from a large collection of documents. These campaigns used to evaluate the relevance of answers given by systems counting how many right answers each system gives. This is called *black-box evaluation* : it is exclusively based on global results.
However, these campaigns do not evaluate the relevance of the criteria the different systems use, neither the contribution of each component. An evaluation of each components, a *glass-box evaluation*, is necessary for improving systems.
In this paper, we will present a definition of what is a question-answering system in section 2, then the tools we developed for evaluating the criteria used by such a system in section 3 and finally we illustrate the process of evaluating some criteria in section 4.

[1] http://trec.nist.gov/
[2] http://clef-qa.itc.it/
[3] http://www.slt.atr.jp/CLQA/

2 Architecture of question-answering systems

A question-answering system allows a user to ask a question in natural language (not with keywords) and provides a precise answer. For example the system must answer *four* to the question *How many were the Beatles?*
Our QA systems is composed of four components : question analysis, document search and analysis, passage selection and answer extraction, which is a classical architecture for QA systems. We will describe the role of each component to explain what we have done for their evaluation.
Question analysis extracts information about the question, which allow the other components to operate in the way that is supposed to. If the criteria are inexact, the possibility of extracting a good answer is reduced. Our system extracts the category of the question (definition, instance...), the semantic type (answers hyperonym) and the focus (entity on which the question is asked).
The second module takes the terms of the question and searches documents where these terms, and their linguistics variations, are present.
The third one consists in selecting sentences which may contain the answer. The sentences are weighted according to their similarity with the question words.
The last module extracts the precise answer of the sentences by applying extraction patterns or selecting the named entity which is expected. Extraction patterns are determined compared to the question category and are based on pivot terms : focus, main verb or semantic type.
As a result, evaluating the accuracy of each components needs an access to the intermediary results to estimate their contribution.

3 Tools for precise evaluation

As we said before, glass-box evaluation allows the improvement of systems by noticing the contribution of each module compared to the evaluation of a global result. Some papers discuss the interest of this approach, which is not conflicting with black-box evaluation. The two types of evaluation are complementary : it depends on what you want to evaluate. 'State-of-the-art systems involve processing whose influences and contributions on the final result are not clear and need to be studied' [3].
On the one hand, the greatest part of the approaches about components evaluation is based on the removal (and substitution) of components. This enables the system to test the components : they can study the results they obtained and in which proportion [1] [6]. On the other hand, the Javelin system [5] contains a module for examining the process of the system based on controlling the execution and information flow. Our approach belongs to this second part of work.
Our purpose consists in modifying the intermediate results created by the components and inserting the new results in the process, without modifying the components. We can test the modification resulting from the criteria chosen in the question analysis and study their impact towards the other components.
Figure 1 shows where our evaluation is done in our question-answering system FRASQUES[4].

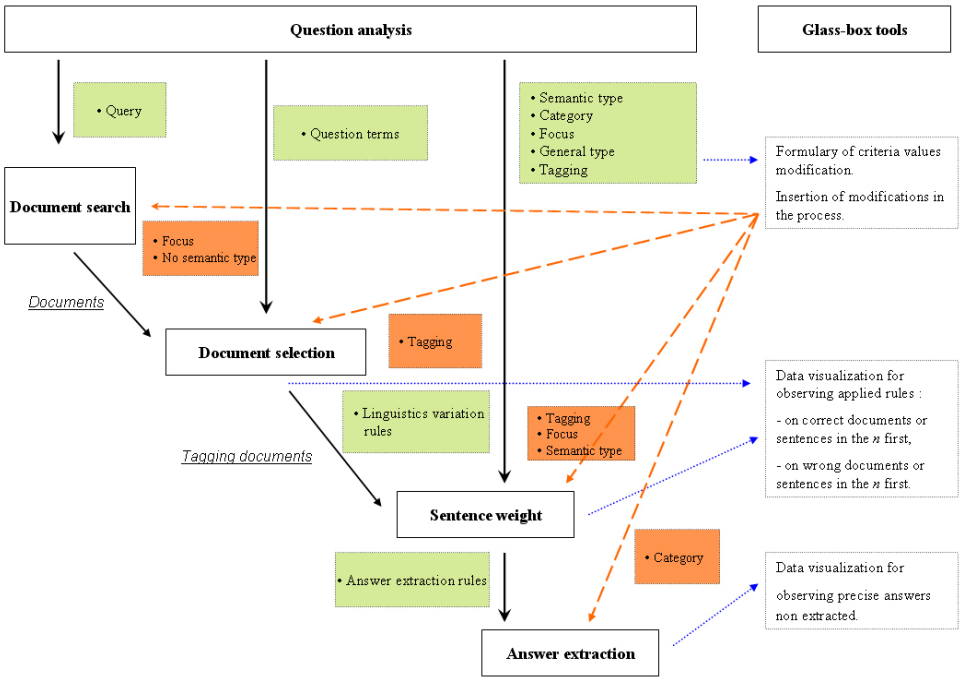

Fig. 1. Glass-box evaluation

Intermediary results of the QA system are generated in files, and all of them are known. By this way, it is possible to store them into a relational database, using XML and XSLT technologies for visualizing request results. These requests either are predefined and can be selected within an interface, or can be written by the user (visualizing how many sentences have a good answer, the instance of a term and its variations,...)
The same interface allows the user to insert new values for some attributes he wants to test. As the different processes of the QA system read their input from files, it is possible to generate a new version of intermediary files that contain these new values. Then, by using a tool that offers the possibility to test the system from certain pre-defined entry points, we can run the system on the new files and study the effect of modifications without modifying the QA system.
The different modules need informations from the question to function. Figure 1 shows which component uses which informations. Our glass-box evaluation tools allow two types of intervention :

- the modification of erroneous information and the insertion of the correct version into the process (dash arrows),
- the precise observation of data (dot arrows).

Consequently, we can test different definitions of the criteria used by the system and evaluate their relevance, and in the same way we can analyse the errors encountered by the system.

4 Examples of glass-box evaluation

4.1 Error analysis

Our interface allows us to analyse the errors of the system and to find the reason of these errors. If the system can find answers in sentences, it has more difficulties to extract them. Thanks to our database, we observed why some extraction pattern did not match. We found three reasons :

- at the question analysis step : bad extraction of the question criteria,
- at the document extraction step : bad tagging of the words,
- at the answers extraction step : non-application of extraction patterns.

After visualizing the errors of execution, we can modify manually the results and mesure again the results we obtain. This analysis of each module put the light on the problems linked to a specific problem, and more precisely which part of this module, is the better way to improve the system. Then a black-box evaluation will confirm the interest of the modifications needed. We will illustrate this method on a criterion.

4.2 Evaluation of a criterion

According to our definition, the focus is the entity about which the question is asked. For example, the question *What year was Martin Luther King murdered?* expects **Martin Luther King** as a focus. This term is the one about which the need of information is required. This information is important for weighting sentences : if a sentence contains this word, there is some chance that the answer is close to this element.
According to our database, we can mesure the relevance of this criterion. It allows us to count the phenomena compared to the questions answered and the questions without answers. This criterion is a good one for us because it reveals when the system works the way it is supposed to. The tables underneath illustrate the impact of the focus on the results.
This evaluation is done on the question set of CLEF05, in the question analysis module. A correct sentence is a sentence extracted by the system which contains the answer, and a wrong sentence is the opposite. A correct focus is a correct extraction of the entity about which the question is answered, and a wrong one consists in the extraction of a wrong word.

Improvement of the focus criterion

Questions	Correct sentences	Wrong sentences	Correct focus	Correct sentences with correct focus
188	148	40	82	49/82

These results refer to sentences which contain the answers (and not to the precise answers extracted from these sentences). We only found 49 correct sentences with the right focus identified by the question analysis. To ameliorate that, we could manually modify this criterion and insert the new results into the process. Then measuring these new results again would enable us to evaluate the relevance of our definition of a focus.
Nevertheless, we can see that the system has problems dealing with the recognition of focus in the answer extraction module : it failed to extract it correctly more than half of the time. Studying the presence of the focus will therefore be an effective means for judging its relevance for the system.

5 Conclusion

The relevance of glass-box evaluation for complex systems is a reality. That aside, it is not easy to create a methodology for evaluating the contributions of the components of any question-answering system. Our method is based on interrupting the flow of the process and modifying it to the consequences of the theories involved in such a complex system. Any question-answering system producing intermediary results would be able to use our glass-box evaluation interface.

References

1. Luis Fernando Costa, Luis Sarmento, Component Evaluation in a Question Answering System. In: Proceedings of the Language Resources and Evaluation Conference (LREC) (2006).
2. Laurence Devillers, Helene Maynard, Patrick Paroubek, Sophie Rosset, The PEACE SLDS understanding evaluation paradigm of the French MEDIA campaign. In: Proceedings of the European Chapter of the Association for Computational Linguistics (EACL) (2003).
3. Laurent Gillard, Patrice Bellot, Marc El-Beze, Question Answering Evaluation Survey. In: Proceedings of the Language Resources and Evaluation Conference (LREC) (2006).
4. Brigitte Grau, Anne-Laure Ligozat, Isabelle Robba, Anne Vilnat et Laura Monceaux, FRASQUES: A Question-Answering System in the EQueR Evaluation Campaign. In: Proceedings of the Language Resources and Evaluation Conference (LREC) (2006).
5. E. Nyberg, T. Mitamura, J. Callan, J. Carbonell, R. Frederking, K. Collins-Thompson, L. Hiyakumoto, Y. Huang, C. Huttenhower, S. Judy, J. Ko, A. Kupse, L. V. Lita, V. Pedro, D. Svoboda and B. Van Durme, The JAVELIN Question-Answering System at TREC 2003 : A Multi-Strategy Approach with Dynamic Planning. In: Proceedings of the Text Retrieval Conference (TREC) (2003).
6. D. Tomas, J. L. Vicedo, M. Saiz, R. Izquierdo, Building an XML framework for Question Answering. In: Proceedings of the Cross Language Evaluation Forum (CLEF) (2005).