A First Step to Address Biography Generation as an Iterative QA Task

Luís Sarmento

LIACC - Faculdade de Engenharia, Universidade do Porto
Rua Dr. Roberto Frias, s/n, 4200-46 Porto, Portugal
las@fe.up.pt

Abstract. The purpose of this article is two-fold: (i) to present RAPOSA, an open-domain automatic question answering system for portuguese that participated in QA track at CLEF 2006 for the first time, and (ii) to explain how RAPOSA is intended to be a key component of a larger information extraction framework that uses question-answering technology as the basis for automatic generation of biographies. We will make a first attempt to classify questions regarding their relevance for iterative biography generation and, according to such classification, we will then explain our motivation for participating in CLEF 2006. We will describe the architecture of RAPOSA, explaining the internal details of each of its composing modules. Next, we will present the results RAPOSA obtained on this year's edition of the QA track, and we will identify and comment on the main causes of error. Finally, we will present directions for future work.

1 Introduction

It is generally accepted that today's QA systems are reasonably good in finding simple answers to several types of questions. For example, for a question like "Who is X ?", with X being a name of a person, most of today's QA systems will easily find at least one "good" answer, usually containing information about the nationality and job description of person X. However, it is highly arguable that such an answer is satisfactory in a realistic scenario: most users would probably expect a more complete biographic profile about person X. In a certain sense, a question such as "Who is X?" can be seen as a series of implied biographic questions. A more user-friendly QA system would provide the user with additional information about X, such as X's place and date of birth, X's place and date of death (if that is the case) or information about X relatives (husband / wife / children / parents...) [1]. Ideally, such QA system would be able to formulate other more specific questions depending on X's profile. For instance, if X was known to be an sportsman, answers to other related questions such as "In which team did X play (in 1993)?" would probably be relevant for the information need initially expressed through the question "Who is X?". Using such an iterative question answering procedure, a system would eventually build a complete biographic profile of X, even if just composed of factoids. Our future goal is to

develop an automatic biography generator system for Portuguese using this iterative question answering strategy. There are several challenges involved in this approach: the issue is not only finding the correct answer(s) to a given question but also being capable of automatic formulating the right biographic-relevant questions.

Our main motivation for participating in QA@CLEF for the first time is (i) to develop a deeper insight regarding biographic questions and (ii) to test our QA system - RAPOSA - in answering such questions. In this year's participation, therefore, we made a first attempt in trying to answer some of those questions with RAPOSA, namely those regarding people (simple definitions), place, dates and quantities.

2 Biography-related Questions

The participation in CLEF helped us to make an initial and informal study about the relevance of certain questions for the purposes of biography generation. We identified three different groups of questions regarding biographical elements.

The first group contains questions that refer to simple attributes common to all people, and that are generally relevant in *all* biographies. Such questions can immediately follow from a "Who is X?" question because they do not require any significant background knowledge or inference about the X person. We will say that these question belong to the *elementary question set*. Looking at the 200 questions set from 2006 QA@CLEF track, we were able to identify 13 question that can be considered part of such an elementary question set (excluding the "Who is X?" questions). The next list shows some examples, along with the question number provided by the organization:

1. Who is Elisabeth The Second's father? (0076)
2. What's the name of George's H. W. Bush's wife? (0105)
3. Which city was Wolfgang Amadeus Mozart born in? (0118)
4. In what year did Charles De Gaulle die? (0179)
5. Where did Kurt Cobain die? (0455)
6. Where does José Rodrigues Filho work? (0760)

There are other question regarding features that, although attributable to all people, may not be equally relevant for all biographies. For example, questions like "How tall is X?", or "What is X weight?", although applicable to any person, are definitely more relevant for a biography of a sportsman than they are for a biography of politician or a writer. On the other hand, from a biographical point of view it seems more natural to ask "Which party does X belong to?" if X is known to be a politician rather than if X is a sportsman, despite the fact that any person can have political affiliations. Such questions are only relevant for the biography of people belonging to certain profile types (e.g. artists, politicians, sportsmen, businessmen) so, for our purposes, we will name these questions as *profile dependent questions*. In 2006, the QA@CLEF question set has several question that could be considered profile dependent questions, such as:

1. When did the official coronation of Elizabeth The Second take place? (0086)
2. What motion picture studio was Cedric Gibbons chief art director of? (0356)
3. Who was the Prime Minister of England before John Major? (0416)
4. Which party does Mahfoudh Nahnah belong to? (0469)
5. Name a film directed by Claude Lanzmann. (0481)
6. What team does Johann Cruyff coach? (0756)
7. What does Vital do Rego teach? (0779)

As it can be immediately seen, the formulation of profile dependent questions involves a certain amount of background knowledge, and some inference over basic information about a person (e.g.: a job description). Such questions may focus, for example, on events or organizations related to the person's profile, or even to other people sharing a similar profile. For example, question 0416 from the previous list is important for the biography of John Major, since we might want to know who preceded him in is job, a fact that is quite relevant when the job is being a Prime-Minister.

There are still other questions that may involve some fruitful *speculation* regarding a given profile. It might be interesting for the QA system to assume some hypotheses and then generate questions based on those (unconfirmed) assumptions. For example, asking "When did X commit suicide?" can be formulated even if it is not confirmed that X did in fact commit suicide. However, if an appropriate answer is found to this question, two very important pieces of information are gathered at the same time. The CLEF 2006 question set has some examples such as "What year was Martin Luther King murdered?" (0114) or "When did Cleopatra commit suicide?" (0799). In some cases, if we take into account a known profile, these assumptions can be even more speculative, but still reasonable. For example, the following questions could be formulated as part of an exploratory strategy for finding information, based on reasonable assumption about the profile type: "Which symphony was composed by Beethoven in 1824?" (0063), "Where did Braque and Picasso work together?" (0348), "How many countries did Nixon visit from 1953 to 1959?" (0390) or "Where did Abba win the Eurovision Song Contest?" (0733). For our purposes, we consider these questions as belonging to the *speculative question set*.

3 RAPOSA

RAPOSA (Respondedor Automático a Perguntas que é Ouro Sobre Azul) is an open domain question answering system for Portuguese. Differently to other question answering systems for Portuguese that make use of extensive linguistic resources [2] or deep parsing techniques [3], RAPOSA uses shallow parsing techniques over the semantic annotation produced by a named entity recognition (NER) system, SIEMÊS [4]. Generally speaking, RAPOSA assumes that the correct answer for several types of question is one entity tagged by SIEMÊS, and its job is to select the right one. The importance of named-entity recognition in QA systems is widely accepted. According to [5], 88.5% of the Portuguese questions in QA@CLEF 2004 refer explicitly to named entities, while 60.5% of the answers

are either named entities or they include named entities. In fact, using a named-entity recognition system as a the major source of semantic information during answering extraction is a standard strategy in question answering systems (as evidenced in [6], [7], [8] or [9]).

RAPOSA is currently a pipeline consisting of 6 modules, and is the result of improvements made to the initial architecture presented in [10]. Each module will be described in turn.

3.1 Question Parser

The Question Parser operates in two steps. First, it invokes SIEMÊS in order to identify named entities in the question: people, places, organizations, dates, titles (e.g.: book titles). These are usually either the arguments of the question to be parsed or some important restrictions. SIEMÊS also helps to find other important elements, such as job descriptions, that may be relevant for parsing the question. In a second step, the Question Parser uses a set of 25 rules to analyze the type of question and to identify its elements. After the type of question has been found, these rules try to identify the arguments of the question, argument modifiers, temporal restrictions and other relevant keywords that may represent good clues for retrieving snippets from the answer collection. Depending of the type of question, the Question Parser also generates a list of admissible answer types that are compatible with the tagging capabilities of SIEMÊS. The output of this module is an object containing the parsed question in a canonical form. The next list illustrates such output, including the tags produced by SIEMÊS:

- Question: Quem foi o último presidente da Rússia antes de 1990 ?
- Question Type: Definition
- Class: Job description
- Question head: “Quem foi o”
- Arguments: “presidente” < job description >; “Rússia” < org >
- Modifiers: “último” < time rest >
- Restrictions: “antes de 1990” < period >
- Answer Type: < human >

Currently, these rules only address some types of questions, namely those that refer to people (”Quem...?” / “Who...?”), to places (”Onde...?” / “Where...?”), to time (”Quando...?” / “When...?” or “Em que ano...?” / “In what year..?”) and to quantities (”Quanto...?” / “How many...?”).

3.2 Query Generator

The Query Generator module is responsible for preparing a set of query objects from the previously parsed question so that the following Snippet Searcher module can invoke the required search engines. The Query Generator selects which words must necessarily occur in target text snippets, and which words are optional. The Query Generator may optionally perform a “dumb” stemming

procedure over all words, except the words the belong to the arguments, to produce a less restrictive query. This helps to increase the number of snippets that match that query, and hopefully improve global recall of RAPOSA (we were not yet able to assess the impact of this option in the global performance of the system). Currently, our stemming procedure is very basic: it simply replaces the last 2 or 3 characters of words to be "stemmed" for a wild-card.

3.3 Snippet Searcher

The Snippet Searcher is the module responsible for obtaining text snippets from which an answer may be looked for, using the query objects given by the Query Generator. The Snippet Searcher is intended to be the interface to web search engines and other text repositories available. Currently, the Snippet Searcher is only using two text databases: the CLEF document collection and BACO [11], a snapshot of the Portuguese web in 2003 [12].

The Snippet Searcher receives a set of query objects and transforms them into lower level search expressions. When using the CLEF document collection or BACO, the Snippet Searcher generates the appropriate SQL statement, since both collections were indexed using the MySQL database engine (text was indexed at the sentence level). Snippets obtained from the CLEF collection and from BACO range from 1 to 3 sentences. Each snippet keeps reference to its source, in order to justify the answers with the document ID (or URL) from which the snippet was extracted. For the QA@CLEF track, only the CLEF document collection was used for obtaining one sentence snippets. After retrieving the text snippets, the Snippet Searcher invokes SIEMÊS to NER annotate them.

3.4 Answer Extractor

The Answer Extractor takes as input the question in canonical form and the list of NER annotated snippets given by the Snippet Searcher. It assumes that the answer to the question is one of the elements tagged by SIEMÊS, and since the type of admissible answers for the question at stake has already been determined by the Question Parser, the answer is usually already quite constrained to the set of *semantically compatible* named entities (or other tagged elements) found in the snippets.

The Answer Extractor has two possible strategies to find candidate answers. The first one is based on a set of *context evaluation rules*. For each tagged text snippet, RAPOSA tries to match certain contexts around the position of the argument (note that the argument of the question must be present in the text snippets). For example, for a question like "Who is X?" with X being the name of a person, the answer is expected to be a job description so that checking the existence of those elements around occurrences of X in certain contexts might lead to the candidate answer. In this case, the rule might check for patterns like "... < job description > X ..." or "... X, < job description > ..." with < job description > standing for the element in the text snippet tagged by SIEMÊS as a job description. RAPOSA has currently 25 of such rules for dealing with

questions of the type "Who is < job description >?" and 6 rules to deal with questions of the type "Who is < person name >?". For RAPOSA's participation in CLEF 2006 we were not able to develop similar rules to deal with other types of questions.

The second strategy available for answer extraction is much simpler: it extracts as possible answer any element tagged with a semantic category that is compatible with the expected semantic category for the question at stake. For example, for a question like "When was < EVENT > ?" the Answer Extractor will collect all elements tagged as < date > in the text snippets (which match the string < EVENT >) provided by the Snippet Searcher. Although this strategy is potentially very noisy, since more than one compatible element may exist in the snippet, one expects the correct answer to be one of the most frequent candidates extracted, provided that there is enough redundancy in the answer collection. We call this the *simple type checking strategy*.

The output of the Answer Extractor, no matter which strategy is used, is a list of answer objects containing the candidate answer and the text snippets from which the answer was extracted. If no answer is found in the text snippets given by the Snippet Searcher, RAPOSA produces a NIL answer for the question, assigning a low value of confidence to the answer (0.25) to acknowledge the fact that such result may be due to a lack of better analysis capabilities.

3.5 Answer Fusion

The role of the answer fusion module is to cluster lexically different but semantically equivalent (or overlapping) answers in to a single "answer group". At this moment, this module is not yet developed: it simply outputs what it receives from the Answer Extractor.

3.6 Answer Selector

The job of the Answer Selector is (i) to choose one of the candidate answers produced by the Answer Fusion module, (ii) to select the best supporting text snippets and (iii) to assign a confidence value to that answer.

Currently, for deciding which of the candidates is the best answer, the Answer Selector calculates the number of supporting snippets for each candidate and chooses the one with the highest number of *different* supporting snippets. The assumption behind this strategy is that by promoting candidates that are supported by many *different* snippets, we eliminate the bias produced by the presence of duplicate documents in the search collections. However, we need to further study the impact of this particular option on the global performance of RAPOSA, comparing it with results obtained when directly using the absolute frequency of the snippets.

Choosing the "best" supporting snippets is quite straightforward when the Answer Extractor uses the context matching strategy because the snippets found must match very specific patterns, which almost always have explicit information for supporting the answer. However, if the candidate answers are obtained using

the *simple type checking strategy*, the Answer Selector has no way, at the moment, of deciding if a given snippet (containing both the argument and the chosen candidate answer) really supports the answer. So, in both cases, the procedure is simply to randomly choose up to 10 different supporting snippets associated with the answer.

For the participation in QA@CLEF 2006, RAPOSA used a very simple rule for assigning the confidence level to answer. The chosen answer - which has at least one supporting snippet - was given a confidence level of 1.0. This value obviously disregards important information such as the number of alternative candidates available and the corresponding number of supporting snippets. After the CLEF event, we made a small improvement in this method, which nevertheless still ignores many important factors. RAPOSA now calculates the confidence level $c(a_i)$ for a given the answer a_i chosen from the set of all candidate answers $A = a_1, a_2, ...a_n$ using Equation 1. This formula takes into account the number of supporting snippets found and, indirectly, the number of alternative answers.

$$c(a_i) = \frac{\#\ snippets(a_i)}{\sum_{k=1}^{n} \#\ snippets(a_k)} \tag{1}$$

4 RAPOSA at CLEF06

We submitted two runs with two different RAPOSA configurations. It is important to say that during our initial study of this problem we believed that the definition questions - "Who is 'person name' ?" and "Who is 'job description' ?" - were the most important ones because they would allow us to determine (or check) the profile type of a given person. Therefore, during the earlier stages of the development of RAPOSA, we were mainly considering these types of questions, and most of the question parsing and answer extraction rules developed for QA@CLEF 2006 had these questions in mind.

The first run submitted, R1, was configured to extract candidate answers using the *context matching rules* developed, i.e. using the most restrictive (and hopefully the highest precision) strategy of the Answer Extractor. However, these rules only covered the 34 "Who is ...?" questions, among the 200 questions that were proposed. All other questions were automatically given a NIL answer with a confidence value of 0, to indicate that RAPOSA did not even try to answer them. Since it was quite disappointing to see RAPOSA ignoring over 80% of the questions (we initially expected the number of "Who" questions would be higher), we made a second attempt to answer other biography-relevant questions, namely those regarding dates, locations and quantities. As it was not possible to develop *context matching rules* for all those cases, we decided to use the more relaxed *simple type checking strategy* to extract the answers for those questions.

Therefore in the second run, R2, RAPOSA was configured to use two alternative strategies to extract answers, chosen according to the type of the question at stake. Thus, for the 34 "Who is...?" questions, RAPOSA was configured to behave exactly as in R1, and perform answer extraction using the same *context*

matching rules. For the other questions - place ("Onde...?" / "Where...?"), time ("Quando...?" / "When...?" or "Em que ano...?" / "In what year..?") and quantities ("Quanto...?" / "How many...?") questions - RAPOSA was configured to extract answers using the *simple type checking strategy*. Again, all other questions would immediately receive a NIL answer with the corresponding value of confidence set to 0. In R2, RAPOSA was able to answer 40 more questions than in R1, making a total of 74 questions.

We have manually checked all the answers against the list of correct answers provided by the organization, considering an answer correct if and only if it exactly matches the answer provided by the organization. We make the distinction between three types of cases among the questions that RAPOSA tried to answer (their confidence values helped to identify the cases):

1. ANS: questions that were answered and supported.
2. NIL1: questions to which RAPOSA did not find any answer after analyzing snippets provided by the Snippet Searcher (NIL + confidence level = 0.25).
3. NIL2: questions to which RAPOSA was not able to find any snippet in the document collection to search for the answer (NIL + confidence level = 1.0).

The results of both runs are given in Table 1, along with the corresponding values of precision *per answer type* ($P_{type} = right_{type}/total_{type}$) and global precision (34 questions in R1, and 74 questions in R2). The last group of columns presents explicitly the performance of the type checking strategy used to answer place, time and quantity questions in run R2 (i.e. those questions not answered in R1).

The precision of R1 is not as high as it was initially expected: a global value 0.18% in answering "Who is ...?" questions seems disappointing. Looking in more detail at the 11 wrong answers produced by RAPOSA in run R1, we observe that most of the incorrect answers have only one supporting snippet. In two of the cases the problem comes from incorrect post-processing of the NER annotation produced by SIEMÊS. For example, in one case the correct answer was "primeiro imperador da China" and SIEMÊS only chunked "imperador da China", which is correct from a NER point of view but is not enough for the answer. In such situations, RAPOSA would need to extract the additional information from SIEMÊS output, but it is not yet able to do it. In two other

Table 1. Results of runs R1 and R2, with the corresponding number of questions where the context matching rules (cmr) or the simple type checking strategy (stcs) were applied.

	R1 = 34 cmr				R2 = 34 cmr + 40 stcs				R2 - R1 = 40 stcs			
type	right	wrong	total	P_{type}	right	wrong	total	P_{type}	right	wrong	total	P_{type}
ANS	5	11	16	0.31	10	24	34	0.29	5	13	18	0.28
NUL1	1	14	15	0.07	7	29	36	0.19	6	15	21	0.29
NUL2	0	3	3	0	0	4	4	0	0	1	1	0
global	6	28	34	0.18	17	57	74	0.23	11	29	40	0.28

cases, the problem came from the inability of RAPOSA to choose the "best" answer from the set of candidates generated, all of them supported by the same number of snippets (only one). In some other cases, the supporting snippets obtained (again usually only one) are only slightly related to the topic of the question, and are actually misleading for the context analysis rules, which are still very naive. One of the most important causes for this problem is related to the rudimentary stemming procedure implemented in the Query Generator. Better query expansion techniques are obviously needed.

The global precision achieved in R2, in which more than 50% of the questions attempted were addressed using the *simple type checking strategy*, was 0.23. If we analyze the answers from run R2 in more detail it is possible to see that about 20% of the incorrect answers result from incorrectly choosing the answer candidate among those elements in the snippet that in fact have the correct admissible semantic type. For example, RAPOSA answered "Lisboa" to the question "Onde morreu Kurt Cobain?" / "Where did Kurt Cobain die?" because there was a reference to Seattle and to Lisboa in the snippet extracted. Another very frequent type of error in run R2, which was not so evident in R1, reveals an important weakness of the Snippet Searcher. After querying the database based on keywords, and after tagging the text using SIEMÊS, the Snippet Searcher does not check whether keywords in the sentence match the expected semantic type of the keywords in the query. This may lead to many problematic situations if keywords have frequent homographs. This is especially severe for questions involving places or people because it is very common for people to have surnames that are also place names. A recent study [13] reports that over 30% of the names of people found in a 32k document sample of the Portuguese web contained at least one geographic / place name. This figure alone illustrates the magnitude of this particular problem.

5 Further Work

We still have to go a long way until RAPOSA reaches the performance level required for generating biographies using an iterative question-answering strategy. Future work will necessarily concentrate on improving existing modules of RAPOSA, especially on solving the various problems already identified. The overall performance of RAPOSA will greatly benefit from more efficient query expansion techniques and from a proper answer fusion module, taking into account biography-related issues. Some work will also be done on improving better context matching rules. The focus of our work will continue to be developing efficient means for answering specific biography-related questions.

Acknowledgements. This work was partially supported by grants SFRH / BD / 23590 / 2005 and POSI / PLP / 43931 / 2001 from Fundação para a Ciência e Tecnologia (Portugal), co-financed by POSI.

References

1. Feng, D., Ravichandran, D., Hovy, E.H.: Mining and Re-ranking for Answering Biographical Queries on the Web. In: Proceedings of the Twenty-First National Conference on Artificial Intelligence (AAAI 2006), Boston, Massachusetts, USA, AAAI Press (July 2006) 1283–1288
2. Amaral, C., Figueira, H., Martins, A., Mendes, A., Mendes, P., Pinto, C.: Priberam's question answering system for Portuguese. [14] 410–419
3. Quaresma, P., Rodrigues, I.: A logic programming based approach to the QA@CLEF05 track. [14] 351–360
4. Sarmento, L.: SIEMÊS - a named entity recognizer for Portuguese relying on similarity rules. [15] 90–99
5. Mota, C., Santos, D., Ranchhod, E.: Avaliação de reconhecimento de entidades mencionadas: princípio de AREM. [16] 161–175
6. Srihari, R.K., Li, W.: A Question Answering System Supported by Information Extraction. In: Proceedings of the 6th Conference on Applied Natural Language Processing, San Francisco, CA, USA, Morgan Kaufmann Publishers Inc. (2000) 166–172
7. Moldovan, D.I., Harabagiu, S.M., Girju, R., Morarescu, P., Lacatusu, V.F., Novischi, A., Badulescu, A., Bolohan, O.: LCC Tools for Question Answering. In NIST, ed.: Proceedings of the 11th Text REtrieval Conference (TREC-2002), Gaithersburg, MD (19-22 November 2002) 144–155
8. Aunimo, L., Kuuskoski, R.: Question Answering using Semantic Annotation. [14] 477–487
9. Costa, L.: Esfinge - a modular question answering system for Portuguese. [17]
10. Sarmento, L.: Hunting answers with RAPOSA (FOX). [17]
11. Sarmento, L.: BACO - A large database of text and co-occurrences. In Calzolari, N., Choukri, K., Gangemi, A., Maegaard, B., Mariani, J., Odjik, J., Tapias, D., eds.: Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC'2006), Genoa, Italy (22-28 May 2006) 1787–1790
12. Cardoso, N., Martins, B., Gomes, D., Silva, M.J.: WPT03: a primeira colecção pública proveniente de uma recolha da web portuguesa. [16] 279–288
13. Chaves, M.S., Santos , D.: What Kinds of Geographical Information Are There in the Portuguese Web? [15] 264–267
14. Peters, C., Gey, F., Gonzalo, J., Müeller, H., Jones, G.J., Kluck, M., Magnini, B., de Rijke, M., eds.: Acessing Multilingual information Repositories: 6th Workshop of the Cross-Language Evaluation Forum, CLEF 2005. Vienna, Austria. September, 2005. Lecture Notes in Computer Science, Springer Berlin / Heidelberg (2006)
15. Vieira, R., Quaresma, P., Nunes, M.V., Mamede, N.J., Oliveira, C., Dias, M.C., eds.: 7th Workshop on Computational Processing of Written and Spoken Language (PROPOR'2006). Itatiaia, RJ. May 13-17, 2006., Springer (2006)
16. Santos, D., ed.: Avaliação conjunta: um novo paradigma no processamento computacional da língua portuguesa. IST Press (2007)
17. Nardi, A., Peters, C., Vicedo, J.L., eds.: Working Notes of the Cross-Language Evalaution Forum Workshop, Alicante, Spain (20-22 September 2006)