A Review on Cooperative Question-Answering Systems

Dora Melo[1], Irene Pimenta Rodrigues[2], and Vitor Beires Nogueira[2]

[1] Iscac, Instituto Politécnico de Coimbra e CENTRIA, Portugal
dmelo@iscac.pt,
[2] Universidade de Évora e CENTRIA, Portugal
{ipr,vbn}@di.uevora.pt

Abstract. The Question-Answering (QA) systems fall in the study area of Information Retrieval (IR) and Natural Language Processing (NLP). Given a set of documents, a QA system tries to obtain the correct answer to the questions posed in Natural Language (NL). Normally, the QA systems comprise three main components: question classification, information retrieval and answer extraction. Question classification plays a major role in QA systems since it classifies questions according to the type in their entities. The techniques of information retrieval are used to obtain and to extract relevant answers in the knowledge domain. Finally, the answer extraction component is an emerging topic in the QA systems. This module basically classifies and validates the candidate answers. In this paper we present an overview of the QA systems, focusing on mature work that is related to cooperative systems and that has got as knowledge domain the Semantic Web (SW). Moreover, we also present our proposal of a cooperative QA for the SW.

Keywords: Question-Answering Systems, Information Retrieval, Information Extraction, Natural Language Processing

1 Introduction

The QA systems try to find answers that are accurate and concise to questions stated in NL, posed by the user in their own terminology [16]. These systems belong to the Computer Science (CS) area and are directly related to the studies in IR and NLP, and fit within the building systems that automatically answer to questions raised by users in NL.

To find an answer to a question, a QA system can resort not only to structured databases but also to sets of documents in NL. The research domain can vary between small sets of documents locally stored, to enterprise internal documents, to networks of news reports, and even to the internet. The main goal of the QA systems is to provide accurate answers to the question posed by users, by consulting its knowledge base.

Research in this area deals with a wide range of question types, including: facts, lists, definitions, hypothetical, semantically limited, language-independent questions (*cross-lingual questions*). Prior knowledge of the type of expected answer help QA systems to extract accurate and correct answers from the collections of documents that make up their knowledge domain.

The first QA systems were developed in 1960's and were essentially NL interfaces for intelligent systems built for specific domains. The advance of the internet reintroduced the need for research techniques pleasing to the user that reduce information overload, posing new challenges for research in automating question answering.

The amount of information on the internet has increased exponentially over the years, with content covering almost any subject. As a result, when users look for certain information, get a little confused with the vast amount of information returned by search engines. Virtually any type of information is available in the internet in one way or another, having billions of web pages available on the internet. Managing such quantities of information is not a simple task. Search engines such as Google and Yahoo, return links along with fragments of text of all documents in response to the request made by users, and that will allow them to navigate the content through a long list of results to look for the answer wanted.

The development of QA systems emerged as an attempt to solve this problem of information overload. QA systems can be classified into two categories according to their domains: closed and open domain. QA systems with closed domain deal with questions based on a specific domain (eg, medicine, music, etc.). These can be seen as simpler systems, since the techniques of NLP can explore specific areas of knowledge, often formalized in ontologies. The specific area of a QA system involves the intensified use of NLP, formalized through the construction of an ontology of the considered domain. Domains may refer to contexts where a limited type of questions are accepted. Open domain QA systems handle questions about anything, and can only rely on general ontologies and world knowledge. Usually there is more information available from which to extract answers.

The remainder of the paper is structured as follows. In Section 2, we present the proposals in the field of QA system that we consider more relevant for our work, namely the ones targeting cooperation and the semantic web. In Section 3, we present the architecture of a typical QA system. In Section 4, we introduce some characteristics about the methodologies that are used more often in QA. Section 5, enumerates several challenges inherent to the development of these systems. In Section 6, we present some current research topics. In Section 7 we present a general view of our proposal for a cooperative QA systems for the SW developed under the PhD in Informatics. Finally, in Section 8, we establish the final conclusions.

2 State of the Art

The most important QA application areas are information extraction from the entire web, online databases, and inquiries on individual websites. Current QA [1] systems use text documents as their underlying knowledge source and combine various NLP techniques to search for an answer to an user question. In order to provide users with accurate answers, QA systems need to go beyond lexical-syntactic analysis to semantic analysis and processing of texts and knowledge resources. Moreover, QA systems equipped with reasoning capabilities can derive more adequate answers by resorting to knowledge representation and reasoning systems like Description Logic and ontologies. A survey on ontology-based QA is presented in [21]. A study on the usability of NL Interfaces and NL query languages, over ontology-based knowledge, for the end-users is presented in [18]. To that end, the authors present four interfaces that enable different search languages and they present a comparative study of their use. They conclude that users have a clear preference for queries expressed in NL and a small set of expressions composed with some keywords or some formal structures.

Several conferences and workshops have been focusing in aspects of search in QA systems. Starting in 1999, the Text REtrieval Conference (TREC)[3] has invested in a trajectory involving QA systems having as main goal the evaluation of systems answering to factual questions using a set of documents from the TREC corpus. A significant number of systems presented in this evaluations were able to successfully combine IR and NLP techniques. In [2], the authors present a review of QA systems and they compare three main approaches to QA systems based in NLP, in IR and in questions modelling, emphasizing their main differences and the application context that is more adequate to each system.

Cooperative QA is an automated QA in which the system, taking as the starting point an input query, tries to establish a controlled dialogue with its user, i.e, the system collaborate automatically with users to find the information that they are seeking. These systems provide users with additional information, intermediate answers, qualified answers, or alternative queries. One form of cooperative behaviour involves providing associated information that is relevant to a query. Relaxation generalizing a query to capture neighbouring information is a means to obtain possibly relevant information. A cooperative answering system described in [12] uses relaxation to identify automatically new queries that are related to the original query. A study on adapting machine learning techniques defined for information extraction tasks to the slightly different task of answer extraction in QA systems is presented in [17]. The authors identified the specificities of

[3] http://trec.nist.gov/

the systems and also tested and compared three algorithms, assuming an increasing abstraction of NL texts. In [7], a semantic representation formalism dedicated to cooperative QA system is presented, this system is based in conceptual and lexical structures and represents homogeneously web texts, NL questions and related answers. This author also presents and analyses some of the prerequisites in order to build cooperative answers depending on the resources, the knowledge and the process. In order to enhance cooperative QA systems, in [23] a set of techniques to improve these systems is presented and the potential impact of their use is discussed.

A cooperative answer [10,13] to a NL question is an indirect answer that is more useful to the user then a direct and literal answer. A cooperative answer may explain a failure that has occurred during the results computation and/or suggest related questions in order to continue with the search. When the system can obtain some results, a cooperative answer can supply additional information that was not explicitly required by the user. Cooperative answers fit into the context of QA systems and they were originally motivated by the wish to approximate system user dialogue from a human dialogue. The cooperative answer processing is preferable to usual techniques of answer extraction focusing on the users since: first it humanizes the system; second it enables the use of adapted vocabulary; and finally it allows the introduction of non-solicited information that may interest the user.

There are some examples of works that try to build answers, instead of merely extract and retrieve. In [28], the authors propose a model for a QA system where the system, departing from the user question, tries to establish a controlled dialogue with the user. In the dialogue, the system has its main goal to identify the user question and to suggest new question related with the user initial question. The dialogue controller is based on the concept structure in the knowledge base, in the domain constraints and in conditioning specific rules. In [15] a system prototype is presented, this system returns cooperative answers, corrects missing concepts, it intends to meet the user needs and it uses the database semantic information in order to formulate coherent and informative answers. The main characteristics of lexical strategies that were developed by humans intellect in order to answer questions are presented in [8]. This author also presents how this strategies can be reproduced in the construction of QA systems in particular in intelligent cooperative QA systems. A answer search method to find answers that are in a neighbour of an answer to the user initial question is presented by [14], this method can be used to process answers that can satisfy the user needs and claims.

Advanced reasoning techniques that are used in QA systems raise new challenges to researchers since answers are not just extracted directly from the text or from structured databases, building an answer can evolve several reasoning forms with the goal of generate explained and justified answers. The integrated knowledge representation and reasoning mechanisms enable the systems, for instance, to anticipate an answer to questions that may raise and to solve cases where the answer can not be found in the knowledge base. These systems should identify and explain false assumptions and others conflict types that might be found in a question.

In [26], an approach to cooperative QA systems is presented, using databases as the domain knowledge source. In [6], the author proposes a logic based model used to generate intentional and precise answers in a cooperative QA system. This author in [5] presents an approach to draw logic based QA systems, WEBCOOP, these systems integrates knowledge representation and reasoning techniques in order to generate cooperative answers to NL questions posed on the web. PowerAqua [20] is a multi-ontologies based QA system that given a NL question returns answers that are computed using relevant resources distributed in SW.

3 The Architecture of a Question-Answering System

The typical architecture of a QA system comprise three main and distinct phases: question classification; information retrieval and document processing; and information extraction.

The question classification is the first phase and consists of classifying questions according to a defined type, generates the kind of the expected answer, extracts keywords and reformulates the questions into multiple questions semantically equivalent. Reformulate a question into a number of questions with similar meaning is also known as question expansion and provides the basis for increasing and improving the performance of information retrieval mechanism.

The information retrieval phase is very important for QA systems. If it is not found in any document a correct answer, the continuity of the process in searching for an answer is finished. The fragments precision and classification that are candidates for the answer may also affect system performance, during the information recovery phase.

The extraction of the answer is the final phase of the QA systems, and states the difference between what is considered a QA system and the usual meaning given to a text retrieval system. The answer extraction technology becomes an influential and decisive factor in the QA system to achieve the final results. Thus, the answer extraction technology is also considered a necessary and important module for the QA systems.

4 Methodologies used

The QA systems are directly dependent on a good search in corpus - without documents containing the answer, there is very little that the QA systems can do. So it makes sense that larger sets of documents generally provide better performance on QA systems, unless the domain of the question is orthogonal to the set of documents. The concept of data redundancy in massive collections of documents, such as internet, i.e. small fragments of information that are susceptible to be formulated in many different ways, in different contexts and documents [19], leads to two benefits: by having the right information and appear in many forms, the burden done in QA systems to perform complex techniques of NLP in order to understand the text is smaller; the correct answers can be filtered out of false positives, taking into account that a correct answer may appear more often in documents that incorrect answers.

Most of the QA systems use NL text documents as domain of knowledge. The techniques of NLP are used both for the processing the questions as well as to index or process the text corpus where answers are extracted.

An increasing number of QA systems use the internet as its corpus of text and knowledge. However, many of these tools do not produce a pleasurable, cooperative and informative answer to the user, which in turn employ superficial methods (techniques based on correspondence between words, models, etc.) to produce a list of documents containing the probable answer.

In current QA systems [1], typically, the questions classifier determines the type of question and the type of the expected answer. After the question is analysed, the system normally uses several modules that apply techniques increasingly complex, in a gradually reduced amount of text. Retrieving documents uses search engines to identify the documents or paragraphs in documents collections that are susceptible to contain the correct answer. Subsequently, a filter select small fragments of text that contain strings of the same type than the expected answer. For instance, if the question is “Who invented Penicillin?”, the filter returns the text that contains names of people. Finally, the answer extraction search for more information or tracks in the text that determines whether a candidate answer can really answer the question.

5 Challenges of Developing Question-Answering Systems

The development of QA systems have released several challenges motivated, mostly, by the exponential increase of the information available, the advance in technology and by the demands and requirements of users. Wherefore, it is now possible to enumerate a collection of problems that continue to have full attention among researchers and were initially identified by a group of researchers and presented in [9]:

Question classes - Different types of questions require the use of different strategies to find the answer. Question classes are arranged hierarchically in taxonomies.

Question processing - The same information can be expressed in various ways. A semantic model of question understanding and processing would recognize equivalent questions, regardless of how they are presented. This model would enable the translation of complex questions into a series of simpler questions, would identify ambiguities and treat them in context or by interactive clarification.

Context - Questions are usually asked within a context and answers are provided within that specific context. The context can be used to clarify a question, resolve ambiguities or keep track of an investigation performed through a series of questions. For instance, the question, “Why did Joe Biden visit Iraq in January 2010?” might be asking why Vice President Biden visited and not President Obama, why he went to Iraq and not Portugal or some other country, why he went in January 2010 and not before or after, or what Biden was hoping to accomplish with his visit. If the question is one of a series of related questions, the previous questions and their answers might guide the system on the intentions of the user.

Data sources - Before a question can be answered, it must be known what knowledge sources are available and relevant. If the answer to a question is not present in the data sources, no matter how well the question processing, information retrieval and answer extraction is performed, a correct result will not be obtained.

Answer extraction - Answer extraction depends on the complexity of the question, on the answer type provided by question processing, on the actual data where the answer is searched, on the search method and on the question focus and context.

Answer formulation - The result of a QA system should be presented in a way as natural as possible. For example, when the question classification indicates that the answer type is a name (of a person, organization, etc.), a quantity (size, distance, etc.) or a date, the extraction of a single datum is sufficient. For other cases, the presentation of the answer may require the use of fusion techniques that combine the partial answers from multiple documents.

Real time question answering - There is need for developing QA systems that are capable of extracting answers from large data sets in several seconds, regardless of the complexity of the question, the size and heterogeneity of the data sources or the ambiguity of the question.

Cross-lingual - The ability to answer a question posed in one language using an answer corpus in another language (or even several). This allows users to consult information that they cannot use directly.

Interactive - It is often the case that the information needed is not well captured by a QA system; the question processing part may fail to classify properly the question; or the information needed for extracting and generating the answer is not easily retrieved. In such cases, the questioner might want not only to reformulate the question, but to have a dialogue with the system.

Advanced reasoning - More sophisticated users expect answers that are outside the scope of written texts or structured databases. To upgrade a QA system with such capabilities, it would be necessary to integrate reasoning components operating on a variety of knowledge bases, encoding world knowledge and common-sense reasoning mechanisms, as well as knowledge specific to a variety of domains.

Information clustering - Information clustering for QA systems is a new trend that is originated to increase the accuracy of question answering systems through search space reduction [27].

User profile - The user profile captures data about the user, comprising context data, domain of interest, reasoning schemes frequently used by the user, common information established within different dialogues between the system and the user. The profile may be represented as a predefined template, where each template slot represents a different profile feature.

6 Current Research Topics

In recent years, the QA systems evolved to incorporate additional domains of knowledge [22,4]. For instance, the QA systems have been developed to automatically answer to questions of geospatial and temporal context, questions of terminology and definitions, biographical questions, cross-lingual questions, and questions about audio content, images or even video. The current research topics of QA system include:

- Cooperation and clarification of questions and/or answers
- Answers reuse
- Knowledge representation and reasoning
- Social media analysis
- Sentiment analysis

7 Cooperative Question-Answering System for Semantic Web

The wide range of challenges related to the development of QA systems, presented above; the growing need for intelligent search engines able to satisfy the demands of many different kinds of Internet users; the need for cooperation and interaction between users and systems; the need to produce, by the system, accurate answers, informative and expressed as closest as possible to NL; were reasons enough to make the decision to proceed with the arduous task: the development of a cooperative QA system for the SW [25,24].

The proposed cooperative QA system receives NL questions and is able to produce a cooperative answer, also expressed in NL, obtained from knowledge base. When the system can not decide the correct path to obtain the answer, it starts a controlled clarifying dialogue with the user. The system includes deep parsing, makes use of ontologies, OWL2 descriptions and other web resources such as WordNet [11] and DBpedia [3].

Our goal is to provide a system that is independent of prior knowledge of the semantic resources by the user and is able to provide a cooperative, direct, accurate and informed answer to questions posed in NL. To this purpose, the architecture of the proposed system is enriched with a Discourse Controller (DC). The DC is invoked after transforming the NL question into its semantic representation and controls all the steps until the end, i.e. until the system can return an answer to the user: from the phase of question classification, passing through the phase of information retrieval, until the phase of answer processing. That is, the DC tries to make sense of the initial question by: analysing the question and the type expected answer; analysing the ontology structure and the structured information available on the web (such as DBpedia); and use the correspondence of similarity between strings and generic lexical resources (such as WordNet), with the objective to provide a clear and informative answer.

The DC deals with the set of discourse entities: verifies the question presupposition, to decide the sources of knowledge to be used; decides when the answer has been achieved or iterates using new sources of knowledge. The decision of when to relax a question in order to justify the answer, when to clarify a question and how to clarify it, is also taken in this module. Thus, the DC represents the intentions and beliefs of the system and the user, the structure of discourse and the context of the question, includes implicit context (such as spatial and temporal knowledge), entities and information useful for the semantic interpretation (like discourse entities used for anaphora resolution, on finding what an instance of an expression is referring to), that allow to add the ability to deal with multiple answers and provide justified answers.

The QA systems strongly depend on reasoning, fact that led us to choose the Logic Programming, specifically Prolog, for their development. Furthermore, there is a vast amount of libraries and extensions for handling and questioning OWL2 ontologies, as well as incorporate the notions of context in the process of reasoning.

8 Conclusion

The main objective of the QA systems is to provide accurate answers to questions posed by users, rather than returning lists of complete documents or fragments of documents that are closer of the expected answer, as with most IR systems. In this paper we presented an overview of the

QA systems, focusing on mature work that is related to cooperative systems and that has got as knowledge domain the SW, highlighting aspects of typical architecture of a QA system, some features on methodologies that are used more often in QA systems, development challenges of QA systems and some current research topics. Finally, we also presented our proposal of a cooperative QA for the SW.

References

1. Allam, A., Haggag, M.: The Question Answering Systems: A Survey. International Journal of Research and Reviews in Information Sciences 2(3), 211–221 (2012)
2. Andrenucci, A., Sneiders, E.: Automated question answering: Review of the main approaches. In: ICITA (1). pp. 514–519. IEEE Computer Society (2005)
3. Auer, S., Bizer, C., Kobilarov, G., Lehmann, J.: Dbpedia: A nucleus for a web of open data. The Semantic Web 4825(Springer), 722–735 (2007)
4. Azzam, S., Humphreys, K.: New Directions in Question Answering. Information Retrieval 9(3), 383–386 (Jun 2006)
5. Benamara, F.: Cooperative question answering in restricted domains: the WEBCOOP experiment. In: Proceedings of the Workshop Question Answering in Restricted Domains, within ACL (2004)
6. Benamara, F.: Generating intensional answers in intelligent question answering systems. Natural Language Generation (2), 11–20 (2004)
7. Benamara, F.: A semantic representation formalism for cooperative question answering systems. In: Proceeding of Knowledge Base Computer Systems (KBCS) (2008)
8. Benamara, F., Saint-Dizier, P.: Lexicalisation strategies in cooperative question-answering systems. In: Proceedings of the 20th international conference on Computational Linguistics. p. 1179. No. Cruse 1986 in COLING '04, Association for Computational Linguistics, Stroudsburg, PA, USA (2004)
9. Burger, J., Cardie, C., Chaudhri, V., Gaizauskas, R., Harabagiu, S., Israel, D., Jacquemin, C., Lin, C.Y., Maiorano, S., Miller, G., Others: Issues, tasks and program structures to roadmap research in question & answering (Q&A). Document Understanding Conferences Roadmapping Documents pp. 1–35 (2001)
10. Corella, F., Lewison, K.: A brief overview of cooperative answering. Journal of Intelligent Information Systems 1(2), 123–157 (Oct 2009)
11. Fellbaum, C.: WordNet: An electronic lexical database. The MIT press (1998)
12. Gaasterland, T.: Cooperative answering through controlled query relaxation. IEEE Expert: Intelligent Systems and Their Applications 12(5), 48–59 (Sep 1997)
13. Gaasterland, T., Godfrey, P., Minker, J.: An overview of cooperative answering. Journal of Intelligent Information Systems 1(2), 123–157 (1992)
14. Gaasterland, T., Godfrey, P.: Relaxation as a platform for cooperative answering. Journal of Intelligent Information 1(3), 293–321 (1992)
15. Gaasterland, T., Godfrey, P., Minker, J., Novik, L.: A cooperative answering system. In: Logic Programming and Automated Reasoning. pp. 478–480. No. X, Springer (1992)
16. Hirschman, L., Gaizauskas, R.: Natural language question answering: The view from here. Natural Language Engineering 7(4), 275–300 (2001)
17. Jousse, F., Tellier, I., Tommasi, M., Marty, P.: Learning to extract answers in question answering: Experimental studies. In: CORIA. p. 85 (2005)
18. Kaufmann, E., Bernstein, A.: Evaluating the usability of natural language query languages and interfaces to Semantic Web knowledge bases. Web Semantics: Science, Services and Agents on the World Wide Web 8(4), 377–393 (Nov 2010)
19. Lin, J.: The Web as a resource for question answering: Perspectives and challenges. In: Proceedings of the Third International Conference on Language Resources and Evaluation. pp. 2120–2127. No. Lrec, Citeseer (2002)
20. Lopez, V., Motta, E.: Poweraqua: Fishing the semantic web. Semantic Web: Research and Applications (2006)
21. Lopez, V., Uren, V., Sabou, M., Motta, E.: Is question answering fit for the semantic web?: a survey. Semantic Web? Interoperability, Usability, Applicability 2(2), 125–155 (September 2011)
22. Maybury, M.: New directions in question answering. Elements pp. 533–558 (2004)
23. Mcguinness, D.L.: Question Answering on the Semantic Web. IEEE Intelligent Systems pp. 6–9 (2004)
24. Melo, D., Rodrigues, I.P., Nogueira, V.B.: Puzzle out the semantic web search. International Journal of Computational Linguistics and Applications 3(1), 91–106 (June 2012)

25. Melo, D., Rodrigues, I.P., Nogueira, V.B.: Work out the semantic web search: The cooperative way. Adv. Artificial Intellegence 2012 (2012)
26. Minker, J.: An overview of cooperative answering in databases. Flexible Query Answering Systems pp. 282–285 (1998)
27. Perera, R.: Ipedagogy: Question answering system based on web information clustering. 2012 IEEE Fourth International Conference on Technology for Education 0, 245–246 (2012)
28. de Sena, G.J., Furtado, A.L.: Towards a cooperative question-answering model. Flexible Query Answering Systems 1495, 354–365 (1998)