Arabic Question Answering with Dialogue Support

By Waheeb Ahmed & Babu Anto P

Kannur University

Abstract- Question Answering (QA) system is a combination of Information Retrieval(IR) and Natural Language Processing (NLP) techniques. It returns a specific answer in response to user question. However, a system that can interact with the user to clarify and refine the answer is required. We propose QA system that adopts a user model for adaptation and a dialogue interface for interaction with the user combined with information retrieval and natural language techniques for Arabic Language. Our system will be able to handle users' questions in natural language and to present answers in in respect to the user's preferences and expected needs. The system achieved a precision of 82.05% and a dialogue success rate of 71.6%. The result is highly promising. As an extension for the present work, we need to make the system more adaptive and capable to learn and evolve with every new interactive scenario.

Keywords: question answering, information retrieval, user modeling, dialogue interfaces, natural language processing.

GJCST-H Classification: *J.4, K.4.2*

ARABICQUESTIONANSWERINGWITHDIALOGUESUPPORT

Strictly as per the compliance and regulations of:

Arabic Question Answering with Dialogue Support

Waheeb Ahmed [α] & Babu Anto P [σ]

Abstract- Question Answering (QA) system is a combination of Information Retrieval(IR) and Natural Language Processing (NLP) techniques. It returns a specific answer in response to user question. However, a system that can interact with the user to clarify and refine the answer is required. We propose QA system that adopts a user model for adaptation and a dialogue interface for interaction with the user combined with information retrieval and natural language techniques for Arabic Language. Our system will be able to handle users' questions in natural language and to present answers in in respect to the user's preferences and expected needs. The system achieved a precision of 82.05% and a dialogue success rate of 71.6%. The result is highly promising. As an extension for the present work, we need to make the system more adaptive and capable to learn and evolve with every new interactive scenario.

Keywords: question answering, information retrieval, user modeling, dialogue interfaces, natural language processing.

I. Introduction

The vast amount of available information on the web is usually accessed using classical search engines. However, the user gets a list of links to documents and it should spend some time searching for the answer in the documents referred by this links. To save users' time QA systems have emerged. The QA system will return a precise/concise answer as a response to the question given by the user instead of a list of links. There are some works which have been introduced in the field of Arabic language. In [1][2] systems which are based on knowledgebase and web are proposed. The systems depend on web as external source of data. A QA system based on big data is proposed in [3]. It finds answers to complex questions from various data sources including structured/unstructured. The system generates variety of semantic structures from these data resources and converts the knowledge extracted into an RDF format. In [4] the issue of question answering in community question answering (CQA) which gathers information from community sites is addressed. In [5] ontology-driven framework is proposed for natural language question answering using user models that are gathered with the help ontology design patterns. Social

Question answering is proposed in [6]. It uses a collaborative paradigm to fulfill complex information needs. JAWEB [7]is a stand-alone a web-based application that receives a question from user and pass it as a query to a search engine to return the answer from the web. Most state-of-the-art question answering systems has the drawback that their output is independent of the user's characteristics, goals and needs. Secondly, most systems are only able to process *factual* questions, i.e. questions regarding dates, quantities, people, numerical, etc. However, there exist some complex/*non-factual* questions which may have multiple answers (e.g. differrent points of view) or complex answers (e.g. an articulated explanation). Such answers should be generated efficiently, enabling the user to understand and clearly differentiate among the different perspectives, portions of an explanation, and so on. Besides, state-of-the-art QA systems lack the interactivity with the user. The traditional question answering session involves the user submitting a question and the system retrieving a result; the session is then ended. In order to overcome these drawbacks of existing QA systems, we propose a question answering system supported with a dialogue interface. The main feature of such system is that it provides output adapted using of a user model and the answer type to be generated, and thus modifies/adjusts both content and presentation of the final answer(s). To the best of our knowledge, this work will be the first on dialogue question answering for open domain.

II. Architecture of QA System

A question is submitted to the dialogue interface, which passes it to the QA system. The question answering module will generate a query from the question terms and pass the expanded query to a search engine and interacts with the user model. The user model gives information on how to adjust the query conditions and how to rank and re-rank the search engine results in accordance with the user level. It is created during the dialogue using information provided by the user and is updated by the dialogue history module. This model supplies information to the dialogue system on the way of interaction and presentation of results to the user.

a) QA Component

The system should be able to differentiate between questions with simple/factual answers and ques-

Author α: B.E. in Computer Science & Engineering from University of Aden, Yemen. email: waheeb2003aden@yahoo.com

Author σ: Ph.D. in Technology from Cochin University of Science & Technology, India, associate professor and working as a research guide at the Department of Information Technology, Kannur University, Kerala, India. e-mail: batop@gmail.com

tions with multiple answers. Each modules per- forms one analysis stage and described in the following sections (Figure 1 shows the architecture)
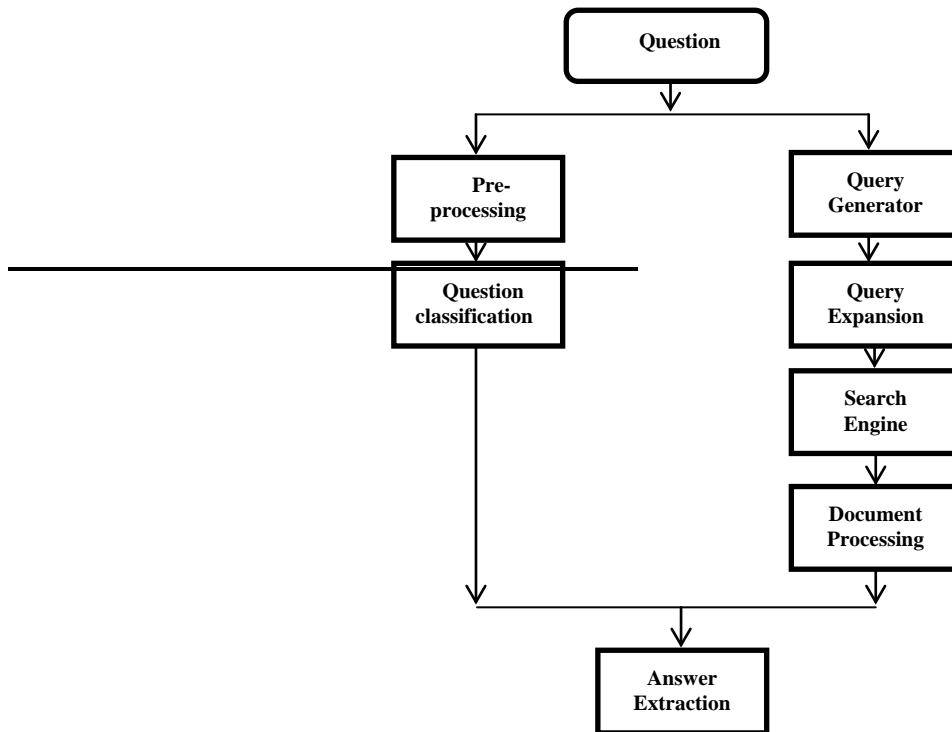

Figure 1: QA component

Figure 1: Architecture of the QA System

i. *Pre-processing*

The question sentence is processed in order to remove diacritics (special Arabic signs on letters) and remove stop words. The stop-words are words that occurs very frequently and does not add any importance to the search process [8][9].

ii. *Question Classification*

The given question is classified to identify the question according to its expected answer type. We used Support Vector Machines from our previous work [10], to help in answer extraction at later stage.

iii. *Query Generation*

A query is generated from the question terms. The query consists of the question terms after removing the stop-words.

iv. *Query Expansion*

The generated query is expanded by adding more terms to it(i.e., synonyms or hypernyms) to improve the search results and stemming can also be used for query expansion[11][12].

v. *Search Engine*

The expanded query is passed to a search engine. In our case, we use Google [13]. The search engine will produce/retrieve a collection of documents. The top ten documents will be extracted.

vi. *Document Processing*

The search results—a list of documents retrieved by their relevance to the generated query— are ranked based on their similarities and to the user model. The documents are divided into passages.

vii. *Answer Extractor*

The analyzed question is compared with the retrieved passages based on similarity metric. A list of answer candidates/sentences is generated which we derive from the relevant passages. In this stage, the ranked passages are filtered based on the user model and answer candidate/sentences are located and prepared for presentation. The passages in the collection are filtered according to their reading difficulty: only those in accordance with the UM's reading level are selected for further analysis.

b) *Dialogue Interface*

The dialogue interface help contribute to the building of a model of the user's interests, goals and level of comprehension/understanding. This consists of

---

[1] All examples are translated into English for this paper.

analyzing the dialogue history which would help the user modeling component to build such a model.

The main aim of the dialogue interface is to provide users with a friendly interface to make their requests. For example, consider the following scenario:

—System: Hi, how may I assist you? 11

—User: I would like to know who is the author of the novel "Treasure Island".

The dialogue component reads the submitted sentence by the user and extracts the following question :"who is the author of the novel Treasure Island?", which will be passed to the question answering module. The dialogue component also help the user disambiguate its queries. For example, assume the user's query is instead:

—User: I would like to know about the novel "Treasure Island".

This question is somehow ambiguous: it is not specified what know about the novel "Treasure Island" in particular the user is intending. In such case, the dialogue component should give a reply like:

—System: Excuse me, did you want to know information about the novel summary or about its author?

—User: Actually, I would like to know about the author of the novel.

In a second stage, the dialogue module has to provide the answer to the user once the question answering component has produced it. The dialogue manager must get consultation of the user model to decide on the most suitable presentation of the answer (i.e. level of readability) and produce the final answer accordingly,

In addition, the dialogue interface should verify if the answer generated was satisfactory and, if not, to carry on the interaction necessary to further clarification/refine or modify the question.

c) User Model

The reason for using user model is that it is an important prerequisites which help the system to adapt to a large range of dialog behavior . The system should consider the user's goals , needs and plans, and also what he/she knows about a domain. Hence, the task to build model of the function of the system in mind is no longer related to the user only. Instead, the system should also to make some inference about what the user needs, believes, and plans. Ideally, the user-system interaction task should not be different from person-to-person communication.

For the purpose of responding in an interactive way to the user, a system must identify the plans formulated in the question provided by the user, store all of these in its knowledge base, test them for embedded hurdles, and generate information to help the user to overcome these hurdles. Thus, a user model is an important requirement for these complex inference procedures.

Another reason for the need of user modeling is that it forms an important prerequisite for intelligent dialog behavior. So, the user modeling is required to identify the elements which the partner of the dialog is talking about, for analysis of the meaning and indirect text indicators in his/her contributions to the dialog.

A User Model contains information that the system uses to identify the knowledge that the user has on the domain, referred as the domain based/ dependent data. The domain based/dependent data are associated with three level functionality namely: Task, Logical and Physical Level. The first level includes the the domain objectives that the user will have to be professional in. In this way, the objectives can be modified based on the progress of the learning process. The second level identifies the user knowledge of the domain and evolves during the user's interaction process. The third level maintains and derives the user knowledge profile. The other type of data is the domain independent data. It consists of two parts: the psychological and the generic models of the user profile. The psychological data are associated with the affective and cognitive aspects of the user. They are more permanent which enable the system to know the features that it must adapt to. The data associated with the user interests, common knowledge and background are saved in the generic model of the user profile. The domain independent data include following elements: Elementary user knowledge, objective and plans, cognitive capacities, references, Learning styles, type and age of user, cognitive style, personality aspects Some of these features are relevant for a specific type of user models and not for others[14].
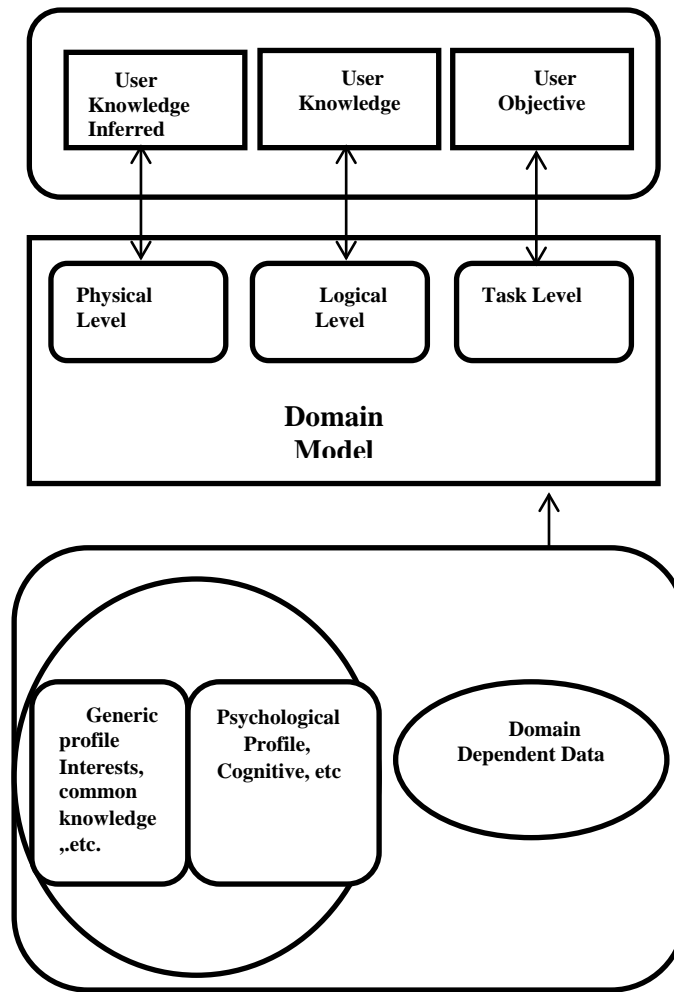
Figure 2: Architecture of the user Model

III. Results and Discussion

For evaluating our system we have used general knowledge questions. we adopt two metrics for evaluating our system: *Precision* and *Dialogue success rate.* The QA system was evaluated by giving 10 sets of dialogue with a total of 50 natural language questions. The two evaluation metrics are defined as follows:

For each set the dialogue success rate =No. of Answers/Responses produced by the system /No. of turns initiated by the user.

Dialogue success rate = (Total of Dialogue success rate for each set / No. of dialogues' sets)*100.

Precision= (No. of correct answers provided by the system/No. of responses generated by the system)*100.

The number of turns issued by the user in a dialogue is the total of the number of questions submitted to the system and the number of responses provided by the user to the system. Each set of dialogue consisted of around 2 to 3 questions. The total dialogue success rate for the 10 sets was obtained as 28.16. The dialogue success rate for the system is calculated as Dialogue success rate= (7.16/10)*100= 71.6%.

Out of 50 questions, system produced answers for 39 questions of which 32 were correct answers. Therefore, the precision of the system is calculated as Precision= (32/39)*100= 82.05%.

The low dialogue success rate is because that the system coverage of the domain is not enough. Another problem is misinterpretation of the dialogue history.

IV. Conclusion

In this paper, we presented a question answering system with dialogue support and a user model component. This system enables the adaptation and clarification of the answer based on the user's level and needs. It scored a precision of 82.05% and a dialogue success rate of 71.6%. The results of the system are promising. As a future direction, the user model needs to be supported with the learning capability to be more flexible.

REFERENCES RÉFÉRENCES REFERENCIAS

1. W.-T. Yih, and H. Ma, "Question Answering with Knowledge Base, Web and Beyond". Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval (SIGIR '16), pp. 1219-1221, 2016..

2. D. Savenkov, and E. Agichtein ,"When a Knowledge Base Is Not Enough: Question Answering over Knowledge Bases with External Text Data." Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval (SIGIR '16), pp. 235-244, 2016.

3. M. Tatu, S. Werner, M. Balakrishna, T. Erekhinskaya, and D . Moldovan, "Semantic Question Answering on Big Data." In Proceedings of the International Workshop on Semantic Big Data (SBD '16), pp.1-6, 2016.

4. K. Zhang, W . Wu, F. Wang, M . Zhou, and Z . Li, " Learning Distributed Representations of Data in Community Question Answering for Question Retrieval." In Proceedings of the Ninth ACM Interna tional Conference on Web Search and Data Mining (WSDM '16), pp. 533-542, 2016.

5. G . Meditskos, S . Dasiopoulou, S . Vrochidis, L . Wanner and I. Kompatsiaris, " Question Answering over Pattern-Based User Models." Proceedings of the 12th International Conference on Semantic Systems (SEMANTiCS 2016), pp. 153-160, 2016.

6. P. Molino, L. M. Aiello, P. Lops , "Social Question Answering: Textual, User, and Network Features for Best Answer Prediction." ACM Transactions on Information Systems (TOIS) Journal, ACM, 2016 October; 35(1):1-40.

7. H . Kurdi, S . Alkhaider, N . Alfaifi. "Development And Evaluation of A Web-Based Question Answering System for Arabic Language". Computer Science & Information Technology (CS & IT). 2014 April; 3(2): 187-202.

8. R. Al-Shalabi, G . Kanan, J.M. Jaam, and E. Hailat ,"Stop-word removal algorithm for Arabic language." Proceedings of the International Conference on Information and Communication Technologies: From Theory to Applications, 2004.

9. W . Ahmed, and B. Anto ,"Question Analysis for Arabic Question Answering Systems". International Journal on Natural Language Computing(IJNLC), 2016 December;5(6):21-30.

10. W . Ahmed, and B. Anto, "Classification of Arabic Questions Using Multinomial Naive Bayes and Support Vector Machines. " International Journal of Latest Trends in Engineering and Technology, Special Issue (SACAIM 2016), 2016, pp. 82-86.

11. H. Al-Chalabi , S. Ray , K. Shaalan . "Semantic Based Query Expansion for Arabic Question Answering Systems." Proceedings of the First International Conference on Arabic Computational Linguistics (ACLing), 2015.

12. S. Khoja , Stemming Arabic Text. [Online]. Available at:http://zeus.cs.pacificu.edu/shereen/research.htm. current access February 2017.

13. Google Search Engine. [Online]. Available at: https://google.com. Current access February 2017.

14. P. Brusilovsky, " Adaptive Hypermedia. "User Modeling and User-Adapted Interaction, 2001; 11: 87-110.