AQuASys: A Question-Answering System For Arabic

SMAN BEKHTI, MARYAM AL-HARBI
Department of computer science
College of computer and information sciences
Al-Imam Muhammad Ibn Saud University
P.O Box 5701 Riyadh 11432
KINGDOM OF SAUDI ARABIA
sbebhti@ccis.imamu.edu.sa, m-alharbi@msn.com

Abstract— In this paper, we present an Arabic Question-Answering system called AQuASys «Arabic Question-Answering System». The proposed system is composed of three modules: A question analysis module, a sentence filtering module and an answer extraction module. The question analysis phase is crucial and plays an essential role in the answer finding phase. Unlike most of the existing Arabic Q-A, our system gives more attention to the question analysis in order to extract, from it, valuable and informative features. These features are decisive for the answer filtering process and consequently have a strong impact on answer finding accuracy performance. The answers relatedness scoring phase serves to find the most accurate answer. Our system attained scores and results are very encouraging. They are presented and discussed in the end of this paper.

Key-Words: - Arabic language, factoid questions, Information retrieval, Morphological analysis, Natural language processing, Question-Answering system.

1 Introduction

Automatic knowledge discovery and information retrieval are becoming more and more essential. This is mainly due to the fact that we are dealing with a huge amount of information especially in the internet. This is true for practically every domain of knowledge.

Access to the relevant information is one of the major problems faced by the user. Particularly, the user lacks time to find a short and precise answer to his/her query among the variety of available documents. Therefore, precision in retrieving the accurate information is crucial and challenging task for Information Retrieval systems developers.

Information Retrieval systems help to retrieve documents relevant to the user's query. In the case of search engines, an enormous list of possibly relevant online documents is returned. Evidently, the user is interested in obtaining a specific and precise answer to a specific question [1]. Therefore, the challenge of developing a system capable of obtaining a relevant and concise answer is obviously of great benefit. For that reason, Question Answering systems are a good solution in dealing with this challenge. They are defined as the answering by computers to precise or arbitrary questions formulated by users.

The challenge becomes greater when we try to automatically process a complex natural language such as Arabic. This complexity is mainly due to the inflectional nature of Arabic. The situation gets worse, in the case of Arabic, when we know that it has a considerable lack of computerized tools and resources in general.

In this paper, we present an Arabic Question-Answering system we called AQuASys «Arabic Question-Answering System». AQuASys is designed to answer fact-based, questions seeking answers related to different types of entities: person, location, organization, time, quantity, etc.

The remaining of this paper is organized as following. In the next section, we give a short overview of Q-A systems with a special attention to the Q-A systems developed for Arabic language. After that, we describe the basis of our system’s analysis that led us to its current design. Next, we discuss the obtained results and achieved scores. Finally, we conclude this work and talk about some perspectives.
2 Related works

As explained in the introduction, Question-Answering systems present a good solution for textual information retrieval and knowledge sharing and discovery. This is reason why a large number of Q-A systems has been developed recently. These systems are designed for numerous world languages, though some languages are better served than others. Moreover, special attention has been given to large scale Q-A techniques for a much larger context like the World Wide Web. Some search engines offer a question answering service with an acceptable performance [2].

When it comes to languages, lot of Q-A systems have been developed in different contexts. In the case of Latin languages, Q-A systems have been extensively studied. English, particularly, is very well served. This is mainly due to the fact that the majority of documents available on the internet are in English [3].

One of the earliest question answering systems is BASE-BALL [4], it supports a finite amount of questions on corpora containing a fixed set of documents. Evidently, Other Q-A systems use the web as a resource. This is the case of START [5] and Swingly [6]. These two systems use search engines to find answers. There is also QALC [7], a Q-A system for English factoid questions in open domain. This system uses a syntactic and semantic analysis for each question. Nevertheless, it presents some errors due to incomplete syntactical rules base. Œdipe is a French Q-A system developed by the LIC2M which applies a set of morpho-syntactical patterns. Then again, it presents a minimalist approach in terms of the used tools [8].

The situation is less bright for Arabic language. Although, it is within the top ten languages in the internet [3], Arabic language lacks computerized tools and resources in general. One kind of these lacking tools are question answering systems specifically designed for Arabic. The developed Arabic Q-A systems are still few compared to those developed for English or French, for instance. This is mainly due to two reasons: lack of accessibility to linguistic resources and tools, such as corpora and basic Arabic NLP tools, and the very complex nature of the language itself (for instance, Arabic is inflectional and not concatenative and there is no capitalization as in the case of English).

Following a comprehensive survey of available Arabic Q-A systems, we realized that there are four well-known such systems:

AQAS: is a knowledge-based Q-A system that extracts answers only from structured data and not from plain text. The developers of AQAS have neither presented nor discussed their system performance and achieved scores [9].

QARAB: A Question Answering System to support the Arabic Language. QARAB is based on IR (Information Retrieval) and NLP (Natural Language Processing) techniques. The developers of QARAB have indicated that they Obtained 97.6% in each of the precision and the coverage [10]. It is noticeable that such accuracy was not achieved in any other language in the Q-A state-of-the-art [8]. The obtained results could be reliable if a test-bed of questions in Arabic were provided in order to allow a comparison between different Q-A systems. QARAB developers adopt a keyword matching strategy, along with matching simple structures extracted from both the question and the candidate documents that are selected by the Information Retrieval (IR) module. They also carry out an analysis of Arabic question forms and attempt to a better understanding of what kinds of answers users find satisfactory [8].

ArabiQA: It deals with factoid questions, integrates NER (Named Entity Recognition) module and adapts JIRS (Java Information Retrieval System) to extract passages from Arabic texts. Yet, Question analysis and Answer extraction modules are not built yet [1].

QASAL: is a semi-automatic Question Answering system for factoid questions using NooJ platform as a linguistic development environment [8]. In addition to the fact that QASAL is not a fully automatic system, in their paper, QASAL developers, present only some examples of questions-answer couples processed by their system. Neither empirical results nor evaluation metrics are presented or discussed to defend the performance of their system.
3 AQuASys System

3.1 Overview

AQuASys is a Question Answering System designed to make it possible for the user to type in a question formulated in Arabic natural language and get an accurate answer to his/her posed question. In particular, AQuASys is designed to answer questions related to a named entity that can be of any types: person, location, organization, time, quantity, etc. Therefore, our system takes, as input, questions starting with interrogative nouns (من من who, ما ما what, أين أين where, متى متى when, كم العددية كم الكمية how many, كم الكمية how much). However and as explained later in this paper, a deep study of sentences formulation and structures in Arabic interrogative forms led us to extend the questioning nouns to a larger number of combined words playing the same role as the interrogative nouns usually considered by the existing Arabic Question Answering systems. We assume that the requested answer is a short passage and does not extend over different documents.

As described in the section evaluation in this paper, the performance of AQuASys is evaluated over a variety of question types provided by Native Arabic users during the testing phases. Also, evaluation metrics are used to assess the accuracy of the returned answers. In addition to integrating a morphological analysis module to our system, we believe that a conducted deep analysis of question forms and structures in natural Arabic language along with the study and the definition of scoring and evaluation rules for the selection of the most accurate answers has a positive impact on the performance of AQuASys as described later in this paper.

3.2 AQuASys architecture

As illustrated in (Fig.1), the global AQuASys architecture is built on four main modules, namely: Question analysis, sentences filtering, candidate answers finding and candidate answers scoring and ranking modules. Each module has been developed based on a number of sub-modules and/or processes.

3.2.1 The Question Analysis Module

The Question analysis phase is crucial and of great impact on the performance of any Q-A system. In AQuASys, the Question Analysis module consists of processing a user question in order to identify, essentially, two important information elements: the type of the expected answer and the key information elements conveyed by the question itself.

![Fig.1 AQuASys global architecture](image)

- The interrogative noun recognition permits, as stated earlier, to identify the expected returned answer. This is essential to identify the relevant sentences that might contain the expected answer.

a) Expected answer type identification

A comprehensive study of the interrogative sentence structures in Arabic language led us to the definition of a number of rules for question types identification and their expected answers recognition. Most of the Arabic Question Answering systems only consider some most known Arabic interrogative nouns, namely, من من who, ما ما what, أين أين where, متى متى when, كم العددية كم الكمية how many, كم الكمية how much). In Arabic, these interrogative nouns can be expressed in other different formulations that have the same interrogative role and meaning. (Table 1) shows a list of types and forms of questions, considered by our system, and their respective expected Answers.

For instance, the interrogative noun (Where أين) which expects a location as an answer, may have other alternative interrogation forms such as (in which town في أي مدينة) or (in which country في أي بلد) and so forth. The situation is similar for the other interrogative nouns. So therefore, some questions may be asked in several possible ways. Considering one question form only and pass over the others definitely affects the performance of any Q-A system.

b) Question segmentation

The second essential task in our Question Analysis module is user’s question segmentation. This task consists of classifying the question words into three categories: interrogative noun, question’s verb and question’s keywords.

Recent Advances in Applied Computer Science and Digital Services

ISBN: 978-1-61804-179-1
- The verb, if there is any, is an important word in any question or sentence, our system identifies the verb within the question. The verb as a special keyword is given more weight in the relevant sentences filtering phase as it is explained later in this paper. The question verb identification is based on questions structures and patterns analysis for Arabic language.

<table>
<thead>
<tr>
<th>Table1. Defined question types and forms</th>
</tr>
</thead>
<tbody>
<tr>
<td>Expected answer type</td>
</tr>
<tr>
<td>-----------------------</td>
</tr>
<tr>
<td>متى (When)</td>
</tr>
<tr>
<td>ما (What)</td>
</tr>
<tr>
<td>أين (Where)</td>
</tr>
<tr>
<td>من (Who)</td>
</tr>
<tr>
<td>كم (How many)</td>
</tr>
<tr>
<td>كم (How many)</td>
</tr>
<tr>
<td>كم (How many)</td>
</tr>
<tr>
<td>كم (How much)</td>
</tr>
<tr>
<td>كم (How much)</td>
</tr>
<tr>
<td>كم (How much)</td>
</tr>
<tr>
<td>كم (How much)</td>
</tr>
<tr>
<td>كم (How much)</td>
</tr>
<tr>
<td>كم (How much)</td>
</tr>
<tr>
<td>كم (How much)</td>
</tr>
<tr>
<td>كم (How much)</td>
</tr>
<tr>
<td>كم (How much)</td>
</tr>
<tr>
<td>كم (How much)</td>
</tr>
<tr>
<td>كم (How much)</td>
</tr>
</tbody>
</table>

- The question keywords are all the words that the user’s question contains other than the interrogative noun and the verb. These keywords are extracted from the formulated question in order to be used in the relevant sentence filtering phase. (Table 2) shows the list of possible Arabic question configurations. We defined and used these combinations for question segmentation and part-of-the-question extraction.

c) Question keywords augmentation
In order to make the relevant sentences retrieving module more accurate, a number of extra keywords are generated and added to the question’s original keywords. These keywords are generated based on the question type.

For instance, if the question is about a location, in the form of: (أين تقع الرياض - Where is Riyadh located), then location related keywords like: موقع, مكان, 133
location, بلد country, مقا عة province, etc. (see Table 2) are generated and added to the list of the question original keywords. The augmented list of keywords is used in the relevant sentence filtering phase to retrieve the candidate answers. The following algorithm fragment is an example of how a “When” question is processed:

ALGORITHM:

1 IF question’s first token = "متى" // "When"
2 THEN the expected answer = time
3 SET verb = second token
4 EXTRACT and BUILD the question keyword list
5 AUGMENT the keyword list with time relevant extra keywords // such as: year, شهر, month, etc.
5 END IF

3.2 The relevant sentences filtering Module

This module is fundamental and consists of identifying the sentences which are the most relevant to the user’s question. It uses all the information extracted by the question analysis module in order to recognize the sentences that most likely contain the relevant answer. These sentences are then ranked by a scoring and ranking module according to their relevance as it is explained later in this paper.

Moreover, after the question is analyzed, the system applies a stemming technique on both the extracted keywords of the question and the documents’ words. This is essential to help overcoming grammatical variations of the words, and then locating the relevant sentences in the document using the string matching approach.

In this phase, the relevant sentences are identified based on the presence of the user’s question keywords in these sentences. These keywords, including the question verb, are considered, in both their actual and stemmed forms, during the documents parsing process. As a preliminary shortlisting, a sentence is identified if it contains at least one of the user’s question keywords (actual or stemmed form). The selected sentences will be filtered upon the number of keywords they contain. In extract keywords stems, we adopted and integrated a stemmer for Arabic language.

a) Integration of a stemmer for Arabic language

Since it is possible that some of the question keywords may not be present in any of the document sentences in their same actual form in the user’s question, the use of a stemmer is evidently of high importance. For instance, if the user inputs the following question: “أين ولد الخوارزمي؟” that is: “Where was Al-Khwarizmi born?” knowing that in a searched document, there is the following sentence (which is the exact answer to the posed question): "الخوارزمي مولود في مدينة خوارزم في خراسان,” meaning “El-Khwarizmi was born in the city of Khwarezm in Khorasan”, and if the stemming mechanism is not used, then this sentence may not be considered as a relevant answer. In fact, The conjugation of the verb (to bear, ولد, was born) in the user’s question, that is (ولد, born) is different from its conjugation in the answer (مولود, born). This is due to the fact that Arabic language has a complex morphology due to its highly inflectional lexicon [11, 12]

Our system uses the stemmer module to extracts, from the user’s question, the following keywords:

ٍ�ٍد [wulid] + خواعزمي [Khwarezmi]

Likewise, once the words, in the relevant sentence above, are stemmed, we get the following:


We can notice that the verbs in the question and the sentence are returned to their root form. Therefore, although their forms appear different in the question and the sentence, the system considers that the two words are similar.

As a stemming algorithm, we adopted Khoja’s stemmer [13]. This stemmer performs the following key tasks [10,13]:

- Removing the Definite Article ال “al”
- Removing the Conjunction Letter و “w”
- Removing Suffixes
- Removing Prefixes
- Pattern Matching

Khoja’s stemmer (Fig. 2) also returns the main root in multiple choices. This gives more possibilities for similar keywords matching.
3.3 Scoring and ranking module
Once the sentences which contain the potentially accurate answer are extracted, our Question-Answering system begins the scoring and ranking procedure (Fig. 3). The potential answers are then displayed to the user ranked according to their respective accuracy rates. The scoring procedure is decisive and permits to identify the sentences that are the most relevant to the user’s question.

![Fig. 3 AQuASys state chart diagram](image)

We developed our accuracy scoring formulas based, essentially, on a study of text similarity measuring techniques [14]. In fact, many studies showed that Question-Answering applications require similarity identification between a question-answer or question-question pairs [15].

<table>
<thead>
<tr>
<th>Type of question</th>
<th>formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>When question (different forms)</td>
<td>Score = Score I + F5 + F6</td>
</tr>
<tr>
<td>Where question (different forms)</td>
<td>Score = Score I + F5 + F7</td>
</tr>
<tr>
<td>What of question</td>
<td>Score = Score I + F5</td>
</tr>
<tr>
<td>How many</td>
<td>Score = Score I + F8</td>
</tr>
<tr>
<td>How long</td>
<td>Score = Score I + F9</td>
</tr>
<tr>
<td>How old</td>
<td>Score = Score I + F10</td>
</tr>
<tr>
<td>Length question</td>
<td>Score = Score I + F10</td>
</tr>
<tr>
<td>Distance question</td>
<td>Score = Score I + F12</td>
</tr>
<tr>
<td>How much money question</td>
<td>Score = Score I + F13</td>
</tr>
<tr>
<td>Weight question</td>
<td>Score = Score I + F14</td>
</tr>
<tr>
<td>Height question</td>
<td>Score = Score I + F15</td>
</tr>
<tr>
<td>Size question</td>
<td>Score = Score I + F16</td>
</tr>
<tr>
<td>Depth question</td>
<td>Score = Score I + F17</td>
</tr>
<tr>
<td>Width question</td>
<td>Score = Score I + F18</td>
</tr>
<tr>
<td>Speed question</td>
<td>Score = Score I + F19</td>
</tr>
<tr>
<td>Temperature question</td>
<td>Score = Score I + F20</td>
</tr>
<tr>
<td>Rate question</td>
<td>Score = Score I + F21</td>
</tr>
<tr>
<td>Surface question</td>
<td>Score = Score I + F22</td>
</tr>
</tbody>
</table>

The scoring process we adopted performs two consecutive scoring tasks:

1. Calculating the initial score using formula (1), bellow. This formula is applied to all types of questions. Formula (1) is based on four important criteria:
   a. Number of existing question keywords in the sentence in their exact actual forms.
   b. Number of the existing question keywords’ stems in the sentence.
   c. Keywords appearance order similarity between the user’s question and the sentence.
d. Keywords direct sequence as it appears in the user’s question.

**Formula (1):**  \[ \text{score I} = F_1 + F_2 + F_3 + F_4 \]

The four considered factors (\( F_i \) / \( 1 \leq i \leq 4 \)) are defined below:

- \( F_1 = \left( \frac{\eta_k}{\theta_k} \right) \times 100 \)

  Where:
  - \( \eta_k \) represents the total number of matching keywords (in their actual forms as they appear in the question) in a candidate sentence.
  - \( \theta_k \) represents the total number of keywords, in their actual forms in the question.

- \( F_2 = \left( \frac{\eta_p}{\theta_p} \right) \times 100 \)

  Where:
  - \( \eta_p \) represents the number of matching keywords’ stems in a candidate sentence.
  - \( \theta_p \) represents the total number of keyword’s stems

- \( F_3 = \left( \frac{\omega_k}{\theta_p - 1} \right) \times 100 \)

  Where:
  - \( \omega_k \) represents the factor of the order of keywords, after being stemmed.
  - \( \theta_p \) represents the total number of keyword’s stems.

The **Order factor** “\( \omega_k \)” is a variable incremented by 1 each time two keywords in a candidate sentence appear in the same order as they do in the user’s question.

- \( F_4 = \left( \frac{\delta_k}{\theta_p - 1} \right) \)

  Where:
  - \( \delta_k \) represents the factor of the sequence of keywords (after being stemmed)
  - \( \theta_p \) represents the total number of keyword’s stems

The **sequence factor** “\( \delta_k \)” is a variable incremented by 1 each time two keywords in a candidate sentence appear in the same direct sequence (with no preceding word(s)) as they do in the user’s question. For instance, in the following question:

"في أي بلد تقع أعلى قمة جبل في العالم؟"

That is: “In which country is the highest mountain peak in the world located?”

The keyword (قمة - peak) position in the question is right after the keyword (أعلى - highest). Likewise, the keyword (جبل - Mountain) position in the question is right after the keyword (قمة - peak) with no separating word. This direct question keywords sequence is taken into consideration, by way of the sequence factor, when searching for the candidate answer(s).

2. The second scoring function consists of calculating the final score of a candidate sentence based on the type of the posed question using the rules shown in (Table 3).

In fact, each scoring factor (\( F_i / 6 \leq i \leq 22 \)) is calculated based on the question’s specific category (Table 4). For instance, if the user asks a question related to a time ("متى/when" question) then, in addition to score I, the system gives extra credit to the sentences containing:

a. The main verb present in the asked question
b. Keywords indicating time, such as: ("عام" - year), ("شهر" - month), ("قرن" - era), etc.
4 Results and discussion

In order to evaluate the performance of AQuASys, we used a corpus (ANERcorp: 150,000 tagged tokens) as well as few gazetteers (ANERgazet) available online [16]. 80 questions of different types were asked to AQuASys. These questions are related to entities of types: person, organization, location, date and number. (Fig. 4) shows an example of an asked question and the returned answers ranked according to their respective scores.

As shown in (Table 5), we obtained an overall recall rate of 97.5% (the recall rate is computed as the number of relevant answers retrieved by the number of relevant answers in the documents) and 66.25% as a precision rate (the number of question answered correctly by the total number of question asked).

The obtained results demonstrate that our system presents a very good performance in terms of recall. In fact, we expected this good recall performance and we believe that this is mainly due to both: questions keywords augmentation method we adopted and the statistical approaches.

![Fig. 4](image_url)
5 Conclusion and future work

We presented, in this paper, AQuASys: an Arabic Question Answering System. This system deals with unformatted questions written in an Arabic natural language. AQuASys is designed to identify and extract the most accurate answer to the user’s question. It also returns a number of additional potential answers ranked by their relevance scores.

In fact, as explained in this paper, AQuASys is composed of a number of complementary fundamental subsystems which have a direct effect on the performance of the whole system. The current scores are very encouraging. Nevertheless, we are keen on improving our system performance, especially the precision degree.

We, particularly, dwell on expanding further the information extracted from the user’s question by including Arabic words cognitive synonyms to the list of the question original keywords (Though such a linguistic resource for Arabic is not available yet). Moreover, we think that an integration of Named Entity Recognition module will definitely boost the system performance. In fact, [18] state that an analysis showed that over 80% out of the 200 questions asked sought a Named Entity as a response.

Furthermore, one of our focal plans is to test our Q-A system on a much larger corpus using a bigger number of diverse questions. This will make the obtained results more credible. Finally, as a long term ambition, we intend to consider studying the processing of the “why” and “how” question types. This kind of questions requires a deeper linguistic and semantic analysis but is necessary if we aim to build a Question-Answering system comprehending all users’ possible questions and queries.

References:


Similarity Measures”. DaWaK 2008: 305-316


