Proceedings Paper:
Abu Shawar, B and Atwell, ES (2009) Arabic question-answering via instance based learning from an FAQ corpus. In: Proceedings of the CL2009 International Conference on Corpus Linguistics. CL2009 International Conference on Corpus Linguistics, 20-23 Jul 2009, University of Liverpool, UK. UCREL, Lancaster University.

Arabic question-answering via instance based learning from an FAQ corpus

Bayan Abu Shawar
Information Technology and Computing Department
Arab Open University
b_shawar@aou.edu.jo

Eric Atwell
School of Computing
Leeds University
eric@comp.leeds.ac.uk

Abstract

In this paper, we describe a way to access Arabic information using chatbot, without the need for sophisticated natural language processing or logical inference. FAQs are Frequently-Asked Questions documents, designed to capture the logical ontology of a given domain. Any Natural Language interface to an FAQ is constrained to reply with the given Answers, so there is no need for NL generation to recreate well-formed answers, or for deep analysis or logical inference to map user input questions onto this logical ontology; simple (but large) set of pattern-template matching rules will suffice. In previous research, this works properly with English and other European languages. In this paper, we try to see how the same chatbot will react in terms of Arabic FAQs. Initial results shows that 93% of answers were correct, but because of a lot of characteristics related to Arabic language, changing Arabic questions into other forms may lead to no answers.

Keywords: chatbot; FAQs; information retrieval; question answering system

1. Introduction

Human computer interfaces are created to facilitate communication between human and computers in a user friendly way. For instances information retrieval systems such as Google, Yahoo, AskJevees are used to remotely access and search a large information system based on keyword matching, and retrieving documents. However, with the tremendous amount of information available via web pages, what user really needs is an answer to his request instead of documents or links to these documents. Form here, the idea of question answering systems raised up to surface. A question answering (QA) system accepts user's question in natural language, then retrieve an answer from its knowledge base rather than "full documents or even best-matching passages as most information retrieval systems currently do." [1]

QA systems are classified into two categories [2]: Open domain QA; and close domain QA. Closed-domain question answering systems answers questions in specific domain such as medicine, or weather forecasting. In contrast, open domain question answering answers questions about everything only and relies on general ontology and world knowledge. In recent years, "the combination of the Web growth and the explosive demand for better information access has motivated the interest in Web-based QA systems" [3].

Katz et al.[4], addressed three challenges face QA developers to provide right answers: "understanding questions, identifying where to find the information, and fetching the information itself". To understand questions and retrieve correct answers, QA systems use

different NLP techniques such as: super vector machine to classify questions, and HMM based named entity recognizer to obtain right answer [5]; Others use a surface patterns to extract important terms from questions, constructs the terms' relations from sentences in the corpus, and then use these relations to filter appropriate answer candidates [6].

In contrast to English and other European languages, Arabic language suffers from shortage in NLP resources and tools. In this paper we will use an Arabic QA corpus to retrieve answers for questions without the need for using sophisticated NLP through using an interface which fools users into thinking that they speak/ask a real human; chatbot.

A chatbot is a conversational software agent, which interacts with users using natural language. The idea of chatbot systems originated in the Massachusetts Institute of Technology [7], where Weizenbaum implemented the Eliza chatbot to emulate a psychotherapist. After that, Colby developed PARRY [8] to simulate a paranoid patient. "Colby regarded PARRY as a tool to study the nature of paranoia, and considered ELIZA as a potential clinical agent who could, within a time-sharing framework, autonomously handle several hundred patients an hour."

Nowadays several chatbots are available online, and are used for different purposes such as: MIA which is a German advisor on opening a bank account; Sanelma a fictional female to talk with in a museum that provides information related to specific piece of art; Cybelle, and AskJevees, a web-based search engine.

The remainder of this paper describes our ALICE/AIML architecture in section 2. The logic of FAQs is presented in section 3. The Arabic WWW FAQs that is used to retrain ALICE with the adapted program is described in sections 4 and 5 consecutively. Results and conclusions are discussed in sections 6 and 7 consecutively.

2. ALICE/AIML chatbot architecture

We have worked with the ALICE open-source chatbot initiative. ALICE[1] is the Artificial Linguistic Internet Computer Entity, originated by Wallace in 1995. A sample of chatting with ALICE is shown in figure 1. ALICE stores knowledge about English conversation patterns in AIML files. AIML, or Artificial Intelligence Mark-up Language, is a derivative of Extensible Mark-up Language (XML). It was developed by the Alicebot free software community during 1995-2000 to enable people to input dialogue pattern knowledge into chatbots based on the ALICE free software technology.

```
User: Hello, my name is George and I am the human.
ALICE: Hi George.
User:  Would you like to play a game?
ALICE: Sure.
User: Ok then. I will email it to you by use of the CD drive.
ALICE: What hobbies do you enjoy?
User: I like cinema.
ALICE: What types of films?
User: All.
ALICE: What movie did you see this year that you liked?
User: I don't have time for movies.
ALICE: I thought you liked cinema.
User: I did, but then you started to say stupid things.
```

Fig. 1. A sample of conversation with ALICE (from Loebner prize contest 2004)

[1] http://www.Alicebot.org/

AIML consists of data objects called AIML objects, which are made up of units called topics and categories as shown in figure 2. The topic is an optional top-level element, it has a name attribute and a set of categories related to that topic. Categories are the basic unit of knowledge in AIML. Each category is a rule for matching an input and converting to an output, and consists of a pattern, which represents the user input, and a template, which implies the ALICE robot answer. The AIML pattern is simple, consisting only of words, spaces, and the wildcard symbols _ and *. The words may consist of letters and numerals, but no other characters. Words are separated by a single space, and the wildcard characters function like words. The pattern language is case invariant. The idea of the pattern matching technique is based on finding the best, longest, pattern match.

```
<aiml version="1.0">
<topic name="the topic">
      <category>
      <pattern>USER INPUT</pattern>
      <that>THAT</that>
      <template>Chatbot answer</template> </category>
      ..
</topic> </aiml>
```

Fig. 2. The AIML format

2.1. Types of ALICE/AIML categories

There are three types of the AIML categories: atomic categories, default categories, and recursive categories.

Atomic categories are those with patterns that do not have wildcard symbols, _ and *, e.g.:

```
<category><pattern>WHAT IS 2 AND 2</pattern>
<template> It is 4 </template></category>
```

In the above category, if the user inputs 'What is 2 and 2', then ALICE answers 'it is 4'.

Default categories are those with patterns having wildcard symbols * or _. The wildcard symbols match any input but they differ in their alphabetical order. Assuming the previous input WHAT IS 2 AND 2, if the robot does not find the previous category with an atomic pattern, then it will try to find a category with a default pattern such as:

```
<category> <pattern>WHAT IS 2 *</pattern>
   <template><random>
       <li>Two.</li>
       <li>Four.</li>
       <li>Six.</li>
   </random></template>
</category>
```

So ALICE will pick a random answer from the list.

Recursive *categories* are those with templates having <srai> and <sr> tags, which refer to simply recursive artificial intelligence, and symbolic reduction. Recursive categories have many applications: symbolic reduction that reduces complex grammatical forms to simpler ones; divide and conquer that splits an input into two or more subparts, and combines the

responses to each; and dealing with synonyms by mapping different ways of saying the same thing to the same reply as the following example:

```
<category><pattern>HALO</pattern>
    <template><srai>Hello</srai></template>
</category>
```

The input is mapped to another form, which has the same meaning.

2.2. ALICE/AIML pattern matching technique

The AIML interpreter tries to match word by word to obtain the longest pattern match, as this is normally the best one. This behavior can be described in terms of the Graphmaster as shown in figure 3. Graphmaster is a set of files and directories, which has a set of nodes called nodemappers and branches representing the first words of all patterns and wildcard symbols. Assume the user input starts with word X and the root of this tree structure is a folder of the file system that contains all patterns and templates; the pattern matching algorithm uses depth first search techniques:

If the folder has a subfolder starting with underscore then turn to, "_/", scan through it to match all words suffixed X, if no match then:
Go back to folder, try to find a subfolder starts with word X, if so turn to "X/", scan for matching the tail of X, if no match then:
Go back to the folder, try to find a subfolder start with star notation, if so, turn to "*/", try all remaining suffixes of input following "X" to see if one match. If no match was found, change directory back to the parent of this folder, and put "X" back on the head of the input. When a match is found, the process stops, and the template that belongs to that category is processed by the interpreter to construct the output.

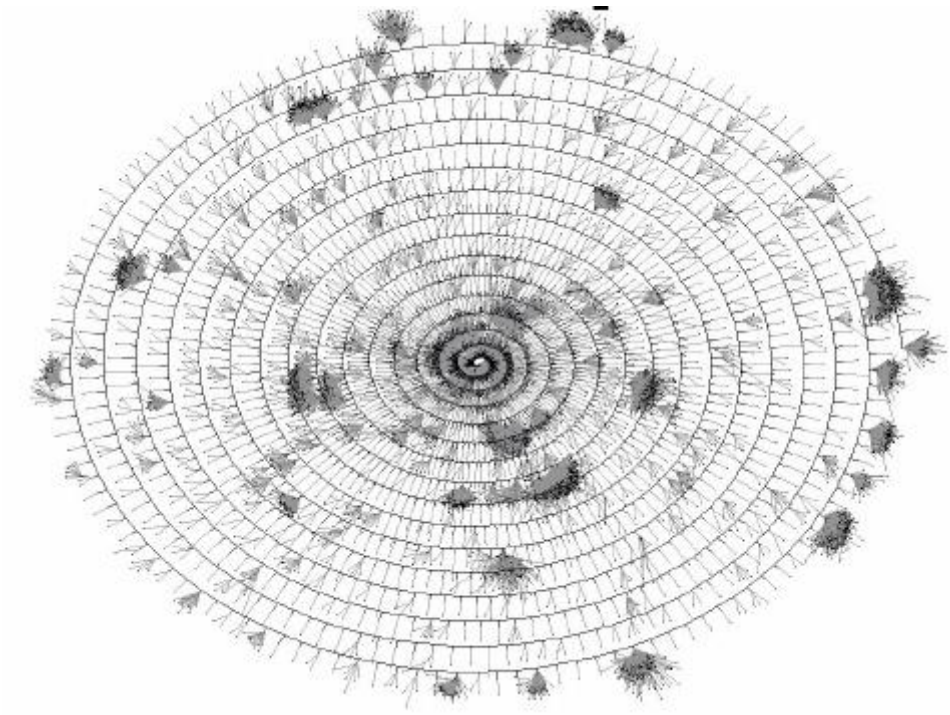

Fig. 1. A Graphmaster that represents ALICE brain

3. The logic of FAQs

We have techniques for developing new ALICE language models, to chat around a specific topic: the techniques involve machine learning from a training corpus of dialogue transcripts, so the resulting chatbot chats in the style of the training corpus [9], [10], [11], [12]. For example, we have a range of different chatbots trained to chat like London teenagers, Afrikaans-speaking South Africans, loudmouth Irishmen, etc by using text transcriptions of conversations by members of these groups. The training corpus is in effect transformed into a large number of categories or pattern-template pairs. User input is used to search the categories extracted from the training corpus for a nearest match, and the corresponding reply is output.

This simplistic approach works best when the user's conversation with the chatbot is likely to be constrained to a specific topic, and this topic is comprehensively covered in the training corpus. This should definitely be the case for a chatbot interface to an FAQ, a Frequently-Asked Questions document. FAQs are on a specific topic, and the author is typically an expert who has had to answer questions on the topic (e.g. helpdesk manager) and wants to comprehensively cover all likely questions so as not to be bothered by these in future. The FAQ is in effect an ontology, "a formal, explicit specification of a shared conceptualization" (Gruber 1993). The "concepts" in this shared conceptualization space are not the Questions but the Answers. The standard "interface" to an FAQ is not a natural-language front end, but just a Table of Contents and/or Index. Users are typically invited to browse the FAQ document till they find the answer to their question; arguably FAQs are really Frequently sought Answers each annotated with a typical Question. Browsing the entire document is fine for limited FAQs, but gets less manageable for larger domains, which may be hierarchically organized. For example, the online FAQ for the Python programming language has several sub-documents for Python subtopics, so users have to navigate a hierarchical ontology.

The logic of chatbot question-answering is built into an FAQ document by the designer. The designer specifies the taxonomy of possible Answers; whatever Question a user may pose, the chatbot can only reply with one or more Answers from this taxonomy, as the topic is comprehensively defined by this ontology. This suggests that sophisticated Natural Language Processing analysis used in systems like AskJeeves is redundant and pointless in an FAQ-query chatbot. Querying an FAQ is more like traditional Information Retrieval: a user query has only to match one or more documents (Answers) in the document set. However, users may prefer to pose a query as a Natural Language question rather than a Google-style list of keywords; so they may yet prefer a chatbot interface to an FAQ over Google-style traditional Information Retrieval.

We adapted our chatbot-training program to the FAQ in the School of Computing (SoC) at University of Leeds, producing the FAQchat system. The replies from FAQchat look like results-pages generated by search engines such as Google, where the outcomes are links to exact or nearest match web pages. However, FAQchat could also give a direct answer, if only one document matched the query; and the algorithm underlying each tool is different.

In the ALICE architecture, the "chatbot engine" and the "language knowledge model" are clearly separated, so that alternative language knowledge models can be plugged and played. Another major difference between the ALICE approach and other chatbot-agents such as AskJeeves is in the deliberate simplicity of the pattern-matching algorithms: whereas AskJeeves uses sophisticated natural language processing techniques including morphosyntactic analysis, parsing, and semantic structural analysis, ALICE relies on a very large number of basic "categories" or rules matching input patterns to output templates.

ALICE goes for size over sophistication: it makes up for lack of morphological, syntactic and semantic NLP modules by having a very large number of simple rules. The default ALICE system comes with about fifty thousand categories, and we have developed larger versions, up to over a million categories or rules.

We tried to see if sophisticated NLP tool, exemplified by AskJevees, is better than keyword-based IR, exemplified by Google, for accessing FAQ. We designed set of typical English-language questions for the School of Computing FAQ, and posed these to both search-engines, constrained to search only the SoC FAQ website. The correct answer was included in 53 percent of AskJeeves answers, and 46 percent of Google answers, which indicated no significant difference in performance: sophisticated NLP is not better than word-based pattern matching when answering questions from restricted-domain FAQ.

However, users may prefer to input natural language questions rather than keywords, so we also asked set of users to compare Google and FAQchat. We set series of information-gathering tasks, and set up an interface which allowed users to type in question; this was sent to both FAQchat and Google, with responses displayed side-byside. We found that 68 percent of FAQchat answers were considered correct by users, but only 46 percent of Google answers were correct. Users were also asked which system they preferred overall: 47 percent preferred FAQchat, while only 11 percent preferred Gooogle. The aim of this evaluation is to show that FAQchat works properly; it is not a search engine, but it could be a tool to access web pages, and giving answers from FAQ databases.

4. Using Web Arabic FAQs to retrain ALICE

The progress in Arabic Natural Language Processing is slower than English and other European languages. Hammo and other researchers [13] referred this to the Arabic languages characteristics which are:

Arabic is highly inflectional and derivational, which makes morphological analysis a very complex task.
The absence of diacritics in the written text creates ambiguity and therefore, complex morphological rules are required to identify the tokens and parse the text.
The writing direction is from right-to-left and some of the characters change their shapes based on their location in the world.
Capitalization is not used in Arabic, which makes it hard to identify proper names, acronyms, and abbreviations.

In 2004, we modified the Java program we developed with the Qur'an, the holy book of Islam [10]. The generated chatbot accepts an Arabic question related to Islamic issues, and the answers are verses from Qur'an that match some keywords. However, because of the Qur'an nature as a monologue text, not as questions and its answers, evaluation for the Qur'an chatbot shows that most of responses were not related to the question. In this paper, we extend our FAQs chatbot systems generated before in English, and Spanish to include Arabic QA.

In this term, we used different Web-pages to build a small corpus consist of 412 Arabic QA, and covers 5 domains:
mothers and pregnancy issues,
teeth care issues[2],
fasting and related issues to health,

[2] D:\ArabicQA_corpora\مركز عالم الابتسامة التخصصي لطب الأسنان وزراعتها - أسئلة متكررة.htm

blood disease such as cholesterol, and diabetes,
blood charity issues[3].

The questions and answers were extracted not from users' forums, but to guarantee its correctness, we gathered it from web pages like medical centers and hospitals.

Different problems raised up that is related to QA format and structural issues which necessitate some manual and automatic treatments as follows:

The questions in these sites were denoted using different symbols: stars, bullet points, numbers and sometimes with ":س" which mean "Q:". To facilitate programming issues, and unify these symbols, all questions were preceded with "Q:" Samples of those questions are presented in table 1.
Another problem was that some of these were in fact PDF files not as web pages, which required to convert it into text ones.
The answers for some questions were long and found in many lines which requires a concatenation procedure to merge these lines together.

Table 1: Samples of questions of Arabic questions

English translation	Arabic question
Q: Why does the wisdom tooth have this name?	س: لماذا سمي ضرس العقل بهذا الاسم ؟
1) What does blood mean?	1) ماهو الدم؟
* What cloths should a pregnant wear?	*ماهي الثياب التي يفضل أن ترتديها الحامل؟

5. Processing the Arabic QA

The Java program that was developed and used before to convert a readable text to the AIML format is adapted to handle the Arabic QA corpus. The program is composed of three sub-programs as follows:

Sub-program 1: Generating the atomic file by reading questions and answers.
Sub-program 2: Constructing the frequency list, and a file of all questions.
Sub-program 3: Generating default files.

5.1. sub-program 1: Generating Atomic file

The first program is generating the atomic file; during this program the following steps are applied:

(1) Reading the questions which are denoted by ":س" ("Q:")
(2) Normalizing the question by: removing punctuations, and un-necessary symbols
(3) Adding the question as a pattern.
(4) Reading the answer which is coming in a separate line after question mark.
(5) Concatenating answer lines till the next question mark found.
(6) Adding the answer as a template.

[3] D:\ArabicQA_corpora\منتدى شباب الخير __ - الاسئلة المتكررة من المتبرعين_.htm

For example: if the Q/A is

What is blood? ماهو الدم ؟

مادة بديعة التركيب تحتوي على خلايا بأنواع مختلفة ، فهناك الكريات البيضاء التي لها أشكال عديدة ، وهناك الكريات - الحمراء التي تمنح الدم لونه ، كما توجد عناصر ضئيلة الحجم تدعى الص ف يحات ، وهناك عوامل عديدة تؤدي لحدوث التخثر وعوامل أخرى تعاكس الأولى ، في الدم يوجد أيضًا مواد عديدة مثل الألبومين والبروتينات والمواد المغذية والأملاح والشوارد ، كما أنه يحمل فضلات ونواتج (التفاعلات التحويلية) التي تتم بالبدن ومواد عديدة أخرى ، وكل ماذكرن اه يوجد ضمن سائل رائع هو المصل ، ومجموع ذلك هو الدم الذي لايدانيه في تكوينه أو وظائفه أي سائل آخر 0

The AIML category will be:

<category>
<pattern> ماهو الدم </pattern>
<template>- مادة بديعة التركيب تحتوي على خلايا بأنواع مختلفة ، فهناك الكريات البيضاء التي لها أشكال عديدة ، وهناك الكريات الحمراء التي تمنح الدم لونه ، كما توجد عناصر ضئيلة الحجم تدعى الص ف يحات ، وهناك عوامل عديدة تؤدي لحدوث التخثر وعوامل أخرى تعاكس الأولى ، في الدم يوجد أيضًا مواد عديدة مثل الألبومين والبروتينات والمواد المغذية والأملاح والشوارد ، كما أنه يحمل فضلات ونواتج (التفاعلات التحويلية) التي تتم بالبدن ومواد عديدة أخرى ، وكل ماذكرن اه يوجد ضمن سائل رائع هو المصل ، ومجموعذلك هو الدم الذي لايدانيه في تكوينه أو وظائفه أي سائل آخر
</template>
</category>

5.2 sub-program 2: Generating the frequency list

The frequency list created using the questions only, since the most significant words will be used within the questions. All questions denoted by <pattern> are read form the atomic file. A file of these questions is generated. After that a tokenization process is applied to have lexical and found its frequencies. As a result a frequency list is created.

5.3 sub-program 3: Generating the default file

(1) Reading the questions and extracting the two most significant words (content words only) which are the least frequent words.
(2) Different categories are added to extend the chance of finding answers as shown below:

Build four categories using the most significant word (least 1) in four positions as patterns and the set of links it has as templates.
Repeat the same process using the second-most significant word (least 2)
Build four categories using the first word and the most significant words (least 1) where the most significant word is handled in four positions.
Build two categories using most significant 1 and most significant 2, keeping the order of position as in the original question.
Build a category using the first word, most significant word 1, and most significant word 2 where the template is a direct answer.

At the end of this stage, two files were generated: an atomic file and a default one. One of the default categories for the above atomic category is:

<category>
<pattern>* الدم</pattern>
<template>- مادة بديعة التركيب تحتوي على خلايا بأنواع مختلفة ، فهناك الكريات البيضاء التي لها أشكال عديدة ، وهناك الكريات الحمراء التي تمنح الدم لونه ، كما توجد عناصر ضئيلة الحجم تدعى الص ف يحات ، وهناك عوامل عديدة

تؤدي لحدوث التخثر وعوامل أخرى تعاكس الأولى ، في الدم يوجد أيضًا مواد عديدة مثل الألبومين والبروتينات والمواد المغذية والأملاح والشوارد ، كما أنه يحمل فضلات ونواتج (التفاعلات التحويلية) التي تتم بالبدن ومواد عديدة أخرى ، وكل ماذكرن اه يوجد ضمن سائل رائع هو المصل ، ومجموعذلك هو الدم الذي لايدانيه في تكوينه أو وظائفه أي سائل آخر
0</template>
</category>

6. Comparing Arabic chatbots with other search engines

Before training ALICE with the generated AIML files, these files were converted into "UTF-8" code to recognize the Arabic letters. For this purpose two steps are taken:

1. All Arabic AIML file are started with: <?xml version="1.0" encoding="UTF-8"?>
2. An online tool was used (Foxe2314) to convert encoding into UTF-8.

As a result five versions of ALICE were generated to cover the five domains as shown in table 2.

Table 2. Online chatbots versions

Chatbot version	Link
Mother&kids issues	http://www.pandorabots.com/pandora/talk?botid=a9b5adeeae36a18a
Blood disease_issues	http://www.pandorabots.com/pandora/talk?botid=e5bab5f0ee36a350
Fasting&health_issues	http://www.pandorabots.com/pandora/talk?botid=d3a868e02e36a18d
Teeth_issues	http://www.pandorabots.com/pandora/talk?botid=ff2870b17e36a14a
Blood_charity_issues	http://www.pandorabots.com/pandora/talk?botid=a555a37e7e36a14d

Table 3 shows the number of categories generated from each WWW FAQs. In total 5,665 categories were generated.

Table 3. AIML categories generated from Arabic WWWFAQs

WWW FAQs	Atomic categories	Default categories
Mother&kids issues	164	2,057
Blood_disease	97	1,292
Blood_charity_issues	69	960
Health&fasting_issues	25	318
Teeth_care_issues	42	614

Fifteen questions were submitted to the generated versions, 93% of answers were correct. Sample of chatting is shown in figure 3.

The same questions were submitted to Google and AskJevees, the recall was 87% for both. However, because Google and AskJevees return documents that hold the answers, we measure how much it is easy to find the answers inside the documents, based on if correct document is the first one in the returned list, and if the answer is found at beginning. In both

[4] http://www.firstobject.com/dn_editor.htm

search engines AskJevees and Google 50% of the answers were found in files, where users need to search again in these file to find their requests.

Human: ماهي أسباب السكري (What are the reasons of diabetes?)
Arabic_Q/A_blood_disease: - نتيجة فشل البنكرياس في إفراز م ا يلزم من الأنسولين يحدث الداء السكري يرتفع عادة بعد تناول الطعام وإجراء التمارين والتعرض للشدة لخفض مستوى سكر الدم الذي.

Human: ما سبب رائحة الفم؟ (What is the reason of mouth smell?)
Teeth_issues: سبب - عوز بعض الفيتامينات 10 سبب هضمي 10% - فموي 80%.

Human: ماهي الثياب التي تلبسها الحامل؟ (What cloths should a pregnant wear?)
Arabic_Q/A_mother&kids_issues: - الحمل عملية وبسيطة وفضفاضة ، لذلك يجب أن تكون الثياب في أثناء ومساعدتها في حدوث الدوالي ، وتجنب استعمال الأحذية ذات الكعب العاليالتي تستبعد المشدات لعوقها الدوران الوريدي التوازن وتسبب آلام الظهر 0 تسيء إلى.

Human: ماهو فائدة الصيام عند مريض السكري (What is the usefulness of fasting for diabetic?)
Arabic_Q/A_fasting&health_issues: يتم خفض نسبة سكر الدم الى الح
الأدنى وتأخذ البنكرياس قسطًا من الراحة خلالالنهار مما يحسن وظيفتها بعد الافطار د.

Fig. 2. A sample of chatting with Arabic QAchatbots

Unfortunately, changing the form of the Arabic question as presented in figure 5 leads to one of two cases:
No answer is generated
Related but not correct answer is obtained.

Table 4. Analysis of answers generated by different Arabic format of the same question

Question attempts	Question Form	Answer is found Y/N	Reason
Original	ماهي الثياب التي يفضل أن **ترتديها** الحامل؟ What cloths should a pregnant wear?	Y	It is the same question as found in corpus, so an atomic match occurs
Form1	ماهي الثياب التي **تلبسها** الحامل؟ What cloths should a pregnant wear?	Y	An Arabic synonym of lexical "wear" is replaced: "**ترتديها** " is replaced with " **تلبسها**". A right answer is returned because the match is generated according to the words "cloths" and word "pregnant" (الحامل, الثياب)
Form2	ماهي ثياب الحامل؟ What are the pregnant cloths?	N	This generates no answer, because the noun "cloths" ("ثياب") is found in the corpus with an article "the" ("ال"), so no match with the least word was found.

In contrast, AskJevees and Google give right answers or related ones even in case the Arabic form of question is changed. There are many reasons which may cause this as listed below:
Arabic nouns and verbs are heavily prefixed. Nouns are usually preceded with the definite article al, and many conjunctions and prepositions are also attached as prefixes to nouns and verbs [14].
Arabic word formation is a complex procedure that is entirely based on root-and-pattern system. A large number of words can be retrieved from one root [14].
Information retrieval is language dependent operation, so retrieving Arabic documents implies retrieving all the variants of search terms using stemmer, morphological analysis, etc, and this is what AskJevees and Google do.
The Arabic chatbots does not apply any NLP techniques; all what a chatbot does is matching with the keywords which were found in the original FAQs without any modification. This was to aim to see how it works without any sophisticate NLP.
Another important reason is that the size of our corpora was small, as a result not a lot of lexical words variants are generated in the frequency list; we believe that if we increase the size of Arabic QA corpora, the possibility of having answers will increase even if the Arabic question form is changed without the need to any NLP techniques.

7. Conclusion

In this paper, we describe a way to access Arabic information using a chatbot, without the need for sophisticated natural language processing or logical inference. FAQs are Frequently-Asked Questions documents, designed to capture the logical ontology of a given domain. Any natural language interface to an FAQ is constrained to reply with the given Answers, so there is no need for deep analysis or logical inference to map user input questions onto this logical ontology. To test this hypothesis, the FAQ in the School of Computing at the University of Leeds was used to retrain the ALICE chatbot system, producing FAQchat. The replies from FAQchat look like results generated by search engines such as Google. As a result of comparison between FAQchat, Google and AskJevees, feedback favorable to FAQchat was gained from almost all users, even those who preferred Google. Using the previous evaluation, we extend our experiment to include Arabic FAQs.

We managed to demonstrate that simple ALICE-style chatbot engine could be used as a tool to access the Arabic WWW FAQs. We did not need sophisticated natural language analysis or logical inference; a simple (but large) set of pattern-template matching rules is sufficient.

8. References

[1] Dumais, S., Banko, M., Brill, E., Lin, J., and Ng, A. (2002). Web question answering: is more always better?. *In Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. (SIGIR 2002).* Tempere, Finland, pp. 291-298.

[2] Kangavari M., Ghandchi S., and Golpour M. A new model for question answering systems. *In proceeding of world academy and science, engineering and technology volume.* 2008. Pp. 536-543

[3] Rosso P., Lyhyaoui A., Penarrubia J., Gomez M., Benajiba Y., and Raissouni N. Arabic-English question answering.

[4] Katz B., Felshin S., Yuret D., Ibrahim A., Lin J., Marton G. Omnibase: Uniform Access to Heterogeneous Data for Question Answering. *Lecture Notes In Computer Science; Vol. 2553. In Proceedings of the 6th International Conference on Applications of Natural Language to Information Systems-Revised Papers.* Pp.: 230 - 234

[5] Zhang D., Lee W. A web-based question answering system. [Online]: http://dspace.mit.edu/handle/1721.1/3693

[6] Cheng-Lung Sung, Cheng-Wei Lee, Hsu-Chun Yen, Wen-Lian Hsu. An alignment-based surface pattern for a question answering system. In proceeding of: This paper appears in: *Information Reuse and Integration, 2008. IRI 2008. IEEE International Conference on.* Las Vegas, 2008, Pp. 172-177

[7] Weizenbaum, J. (1966). ELIZA-A computer program for the study of natural language communication between man and machine, *Communications of the ACM,* Vol. 10, No. 8, pp36-45.

[8] Colby, K. (1999). Human-computer conversation in a cognitive therapy program. In Wilks, Y. (eds.) *Machine conversations*. Kluwer, Boston/Drdrecht/London. Pp. 9-19.

[9] Abu Shawar, B., Atwell, E. (2003). Using the corpus of Spoken Afrikaans to generate an Afrikaans chatbot. *Southern African Linguistics and Applied Language Studies*. Vol. 21, pp. 283-294.

[10] Abu Shawar, B., Atwell, E. (2004). An Arabic chatbot giving answers from the Qur'an. In: Bel, B & Marlien, I (editors) *Proceedings of TALN04*. Vol 2, pp. 197-202 ATALA.

[11] Abu Shawar Bayan, Atwell Eric and Roberts Andy. 2005. FAQchat as an information retrieval system. In: Zygmunt V. (ed.), Human Language Technologies as a Challenge for Computer Science and Linguistics: *Proceedings of the 2ndLanguage and Technology Conference,* Wydawnictwo Poznanskie, Poznan, pp. 274-278.

[12] Abu Shawar, B. Atwell, E. (2005). Using corpora in machin-learning chatbot systems. *International Journal of Corpus Linguistics* 10:4, pp. 489-516

[13] Hammo B., Abu-Salem H., Lytinen S. QARAB: a question answering system to support the Arabic language. *In Proceedings of the ACL-02 workshop on Computational approaches to semitic languages.* 2002. Pp. 1-11

[14] H. Moukdad, Lost in cyberspace: how do search engines handle Arabic queries? *In: Proceedings of the 32nd Annual Conference of the Canadian Association for Information Science*, Winnipeg, 2004 . Available at: www.cais-acsi.ca/proceedings/2004/moukdad_2004.pdf.