Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 2, No. 3, December 2016.

A RULE-BASED APPROACH TO UNDERSTAND QUESTIONS IN ARABIC QUESTION ANSWERING

Emad Al-Shawakfa

CIS Department, Faculty of Information Technology, Yarmouk University,
Irbid 63-211, Kingdom of Jordan
shawakfa@yu.edu.jo

(Received: 30-Jun.-2016, Revised: 08-Sept.-2016, Accepted: 09-Nov.-2016)

ABSTRACT

Research on Arabic Natural Language Processing (NLP) is facing a lot of problems due to language complexity, lack of machine readable resources and lack of interest among Arab researchers. One of the fields that research has started to appear in is the field of Question Answering. Although some research has been done in this area, few have proved to be effective in producing exact relevant answers. One of the issues that affected the accuracy of producing correct answers is proper tagging of entities and proper analysis of a user's question. In this research, a set of 60+ tagging rules, 15+ Question Analysis rules and 20+ Question Patterns were built to enhance the answer generation of Natural Language Questions posed over some corpora collected from different sources. A QA system was built and experiments showed good results with an accuracy of 78%, a recall of 97% and an F-Measure of 87%.

KEYWORDS

Arabic; Question Answering; Question Analysis; Tagging, NLP.

1. INTRODUCTION

After the computer revolution and the rapid spread of computer usage around the world, many researches were conducted in the field of Natural Language Processing (NLP) toward providing a better and easier way of interaction and usage of computers and their applications by different users; especially, naïve ones. For this, we started to see different applications covering many language computing fields like Stemming, Information Retrieval (IR), Information Extraction (IE), Question Answering (QA), Text Classification and Categorization (TC), Machine Translation (MT), among many others. Such research, at the beginning of the era, was mainly conducted using Latin-based languages; especially English, which has formulated the foundation for interfaces with computers.

With the increased volume of information stored in computers; especially databases and recently the Web, people started to look for different ways to help in extracting needed information from different sources of data through expressing their requests in their own daily used natural languages without any technical experience or knowledge. For this, different search engines have started to appear, enabling users to look for information on the web using different Information Retrieval tools.

When one poses a question to a search engine, then based on query words, one might get either too little or too much of documents as a result to his/her query and thus need to dig more into the results

to obtain proper answers he/she is looking for; which might become more time consuming. Obtained results are even sometimes far away from whatever actually needed and hence disappointing. Such disappointment is related mainly to the IR algorithm used by the search engine(s) in retrieving only possible relevant documents; rather than exact answers we are looking for. The concept of Question Answering came as a rescue to solve such a problem.

As defined by [1], Question Answering (QA) is: "the task to automatically providing an answer for a question posed by a human in natural languages". Unlike IR systems, QA systems seek to obtain an Exact Answer to a given question from a list of documents rather than to have an answer being represented as a list of relevant documents; as is the case with search engines. For most of existing QA systems today, results have not always been satisfactory to the users; especially, for the case of Arabic. However, to obtain an exact result of a query, a more detailed process has to be carried on from the point of posing a query to the point of obtaining a requested answer.

Many Question Answering Systems have started to appear in the early 1960s by introducing systems that can be used to extract information from databases using English [2]. Since then, more systems have started to appear, like ([3]-[5]). A recent survey of versatile question answering systems was given by [6]. Research on enhancing the results of QA systems was also developed ([7]-[10]).

Research in the field of Arabic Natural Language Processing (ANLP); especially the field of Question Answering, has lagged behind research in its counterpart Latin-based languages due to many reasons. The complexity of Arabic language itself, the lack of support for Arabic by computers in the early history of computers and the fact that most of the Arabic content on the web at early stages of the digital era was in non-searchable image format are some of such reasons. Furthermore, the lack of standardized machine readable resources, as well as the lack of researchers and/or users who are willing and interested in working with Arabic have made it more difficult to conduct research in the field of Arabic NLP. Regardless of this, many researches in the field of Arabic Question Answering (AQA) have started to appear. Some of the developed systems were AQAS [11], QARAB [12], ArabiQA [13], QArabPro [14], AQuASys [15], QA4MRE@ CLEF [16], among many more.

As the public would say: "Understanding a question is almost half the answer" and since computer systems did not reach that level of intelligence to be able to understand questions properly by themselves, more research in the field of question analysis and understanding are still needed to construct proper answers to posed question(s) by the user(s).

To conduct this research, the researcher has heavily searched the literature for Question Analysis and understanding rules and found out that only very few researchers built a very little number of such rules; a maximum of five rules were found in [14]. In this research and to enhance the accuracy of answer generation, the researcher built more than 15+ detailed Question Analysis rules that are completely different from those available in the literature, combined with 60+ tagging rules and 20+ question patterns. This, as well as the lack of research in Arabic Question Answering, constituted the main contribution and objective behind this research. Since the majority of Arabic QA systems are rule-based, the researcher has adopted this approach in this research as well. Further enhancement of the existing rules as well as building new other rules constitute a future work.

2. RELATED STUDIES

There are different categorizations of QA systems in existence today, depending mainly on what, how and where from the QA system is trying to answer the question(s). Question Answering

systems that are looking only for facts or definitions are called Factoid or Definitional QA Systems [1]. Systems that are looking for a more detailed answer beyond a fact or a definition are called Non-Factoid or Passage Retrieval Systems as also indicated by [17], or even sometimes called List QA Systems [18]. Systems that are looking for a reason or an explanation to some happening, are called Why-QA Systems.

As for the domain being searched, QA systems that deal with limited domains are said to be of Restricted Domain in comparison to Open Domain QA systems that deal with a non-restricted open domain text; like the Internet. Non-restricted open domain QA systems are sometimes referred to as Web-based QA systems. Furthermore, systems might deal with either structured or unstructured sources of data. For structured data, the system would be an interface to a database; as the data would be stored into a database. On the other hand, unstructured data does not have a uniform structure and thus cannot be stored into a database ([18]-[19]). Such systems constitute the majority of today's developed QA systems due to advances in NLP tools that would enable better Information Retrieval capabilities. In addition, there are some types of QA systems that look only for Yes/No answers, like [20]. Other QA systems could also be categorized as either Shallow or Deep QA systems, depending on how much semantic and/or syntactic analysis is being performed to obtain the answer [21].

Another categorization of QA systems is given by [19]. This categorization is based on the methods used for the extraction of the answer. Information Retrieval/Extraction methods would give the first category of QA systems to refer to Web-based and IR/IE-based QA systems. The second category refers to systems that use reasoning in the extraction of the answer. This type of category would refer to Domain-oriented QA systems and Rule-based QA systems. Table 1 gives such characterization of QA System types.

Table 1. Characterization of QA systems [19].

Dimensions	QA Systems based on NLP and IR	QA Systems Reasoning with NLP
Technique	Syntax processing, Named Entity tagging and Information Retrieval	Semantic analysis or high reasoning
Data Resource	Free text documents	Knowledge base
Domain	Domain Independent	Domain-oriented
Responses	Extracted Snippets	Synthesized responses
Questions Dealt with	Mostly Wh- type of questions	Beyond the Wh- type of questions
Evaluations	Use existing Information Retrieval	N/A

Most of Arabic QA systems are of either Factoid or definitional type looking for short answers. Very few have managed to deal with Why and How (much/many/to) types of questions. Examples of such Arabic QA Systems are that of [14], [22] and [18]. Table 2 gives a very good comparison between some of the existing Arabic Question Answering Systems given by [18].

The first Arabic Question Answering System was introduced by [11] and was called AQAS. The AQAS system is a knowledge-based QA system that extracts answers from structured data stored into a Database. The authors of AQAS did not report any testing or evaluation results for their system, so no one could give any advantages or disadvantages of such system.

According to [1], there was no work performed on Arabic Question Answering from 1993 until 2002, when [12] introduced a rule-based Factoid QA system for Arabic called QARAB which deals with unstructured data from documents collected from Al-Raya Newspaper with 113 Factoid

questions fed into the system. However, QARAB did not handle the two types of questions "كيف، لماذا" (How and Why), because they require long and complex processing.

In 2006, [23] introduced an ongoing implementation of a Factoid Arabic QA system that was used for the purposes of QA tracks in the Cross Language Evaluation Forum (CLEF) and Text Retrieval Conference (TREC) competitions. In their paper, the authors have only introduced partially implemented modules of the system in which the Named Entity Recognition (NER) module as well as a Java Information Retrieval System (JIRS) module were embedded. However, this system was completed and introduced later on by [24], in which the effect of correctly identifying a Named Entity (NE) to produce correct answers is emphasized.

Table 2. A comparison between some existing Arabic QA systems ([18]).

Main Features	**QARAB**	**ArabiQA**	**ArQA**	**QASAL**	**AQuASys**	**JAWEB**
Web-based system	×	×	×	×	×	√
Retrieves answers from a corpus	√	√	√	√	√	√
Retrieves answers from the web	×	×	×	×	×	×
Natural language processing tools	√	×	√	√	√	√
Named entity recognition	×	√	√	√	×	×
Answers factoid questions	×	√	√	√	√	√
Answers open domain questions	×	√	×	√	√	√
Supports multiple languages	×	×	×	×	×	×
Supports Arabic language	√	√	√	√	√	√
Provides short answers block	√	√	√	√	√	√
Measures answer precision	-	√	√	-	√	√
Measures answer recall	-	√	√	√	√	√

A Factoid and Definitional Arabic Question Answering system called QASAL was introduced by [25]. The authors have used NooJ platform and local grammars to help in obtaining the right answers. For the Factoid questions, authors have used the collection of Tunisian books as a corpus. However, for the definitional type of questions, the authors used the Arabic version of Google search engine as a web resource to look for Arabic documents with 43 definition questions. According to the authors, 94% accuracy for definitional questions was obtained.

An Arabic QA system (QAS) to answer short Factoid questions in Arabic was described by [26]. The authors based their testing on a collection consisting of 25 manually collected documents that were gathered from the web in addition to some relevant documents that were provided by the authors applying 12 questions to the set. Authors reported different recall levels of {0, 10 and 20%}, where the interpolated precision was equal to 100% and at recall levels 90 and 100% to be equal to 43%. As is the case with QARAB, QAS did not handle the "كيف، لماذا" (How and Why) types of questions due to the complex processing needed.

An Arabic Definition Question Answering system named DefArabicQA was introduced by [27]. This system answers questions of the form "What is X?" with the web as the data source. The authors claim that their system provides effective and exact answers to definition questions expressed in Arabic using little linguistic analysis and language understanding capabilities. To evaluate their system, two experiments were conducted with Google only as a web source in the first experiment and Google coupled with Wikipedia as the web source for the second experiment. The experiments reported a Mean Reciprocal Rank (MRR) score of 0.7 and a question rate of 0.54 for the first experiment and an MRR score of 0.81 and a question rate of 0.64 for the second.

A rule-based Question Answering System for Arabic called QArabPro was developed by [14]. The system performs reading comprehension texts and tries to answer questions posed upon such texts. QArabPro assumes that the answer must exist within one of the documents that were used as a corpus. The authors claim that they have answered all types of questions including the "كم، لماذا" How (much/many) and Why types in contrast to other existing QA systems that avoided such types of questions due to their complexity. To test their system, they used a set of documents that were collected from Wikipedia with 75 documents and 335 questions. According to the authors, the claimed results were of 93% Precision, an 86% Recall and an F-measure of 89%. Obtained test results showed an overall accuracy for "كم" (How much/many) of 69% and 62% for "لماذا" (Why) questions that were handled. However, QArabPro did not handle "كيف" (How) type of questions.

In [15], a Factoid Arabic QA system named AQuASys was developed. A Factoid natural extension to AQuASys with a web interface and an extended corpus was built by [18] which the authors called JAWEB.

A Question Answering System called IDRAAQ was developed in [28] in the framework of the main task of Question Answering for Machine Reading Evaluation (QA4MRE@CLEF2012).This system was based on keywords and structure levels through query expansion and Distance Density N-gram model-based passage retrieval to improve the results of the system. According to the authors, IDRAAQ has obtained promising results with the QA4MRE framework; especially with Factoid type of questions.

In the QA4MRE@CLEF2012 framework, a work on Arabic Question Answering was given by [16]. According to the authors, the work of [16] has obtained an accuracy of 0.19 with very little of reasoning and inference; an issue requested in analyzing and understanding documents for this framework.

An Arabic Language Question Answering Selection In Machines called ALQASIM was introduced in [29]. ALQASIM was used to answer Multiple Choice questions of the QA4MRE. According to the authors, a novel technique was used in understanding and analyzing test documents which led to an accuracy of (0.31) in comparison to accuracies of (0.13) obtained by IDRAAQ [28] and (0.19) by the approach used in [16].

An Entailment-based Why Arabic Question Answering (EWAQ) system was introduced by [30]. According to the author, EWAQ enhanced the accuracy of "Why" questions by improving the re-ranking of passages that are relevant and retrieved by many search engines as possible answers. She claimed that the accuracy of her system has improved over that of search engines; Google, Yahoo and Ask.com.

In most of existing types of QA systems; especially Factoid QA systems, the search will be for a Named Entity as part of an answer; or even the answer itself. In English and other Latin-based languages, it is very easy to locate a noun with all of its categories in a given text due to the capitalization feature that exists in the language itself. However, since Arabic does not support any capitalization of letters and is a highly inflected and derived language with rich morphology and complex syntax, the identification is not straight forward; a process that would be more difficult to carry on for Arabic. For this, a research on handling Named Entities in Arabic was conducted.

Once introduced into research during the sixth Message Understanding Conference (MUC-6), the concept of Named Entity (NE) did not just cover only proper nouns, but also included other types. The types, or classes, that were introduced by MUC-6 for NE were ENAMEX (referring to person names, locations and organizations), NUMEX (referring to money and percentage [numerical] expressions) and TIMEX (referring to time and date expressions).

There are two approaches to build Name Entity Recognition (NER) systems in Arabic; a rule-based approach like that of NERA system was introduced by [31] and a Machine Learning (ML) approach like ANERSys system was built by [24]. Some of the systems that adopted the rule-based approach are: TAGARAB [32], PERA [33], ARNE [34] among many others. As for the ML approach, many researches have been conducted using this approach like those of ([13],[31] and [35]-[37]), among many others. For more information, one can refer to a very good and recent survey of Arabic Named Entity Recognition systems given by [38].

3. THE QUESTION ANSWERING PROCESS

The generic Question Answering System consists of three major modules; namely, the Question Analysis and Understanding Module, the Document/Passage Retrieval Module and the Answer Extraction and Response Generation Module. This research is part of an ongoing research on Arabic Question Answering and is concerned with the first module. Work on other modules of the Arabic Question Answering; IR and Response generation, is currently being carried on.

3.1 Question Analysis and Understanding Module

One of the most important steps in the Question Answering process is the issue of question understanding; a question must be properly analyzed to clarify what is meant by such question thus enabling us to be directed in the proper path of finding the right and exact answer to our query. In this module, a correct understanding of what a question might be looking for; or what is known as the Scope (or Focus) of the Question and Question Type constitute a crucial step toward providing the right answer to a given question.

Regardless of the natural language being used, a question type would fall into one of the following categories:

1) Who/Whose: such type of question will be looking for animate objects such as a person.
2) What/Which: such type of question will be looking for inanimate objects, like an entity or a thing.
3) Where: such type of question usually looks for a place or location.
4) When: such type of question will be looking for a time or time-related information.
5) Why: such questions are usually not easy to answer, but they will be looking for a reason or a cause for the happening of some action.
6) How: depending mainly on what follows; for instance, in the case of How much or How many, such questions will be looking for numbers or quantity. However, if How is not followed by much or many, these questions will be looking for a process or procedure on doing some action.

In Arabic, there are more ways to ask questions than in English. Arabic uses the same WH interrogative nouns of English in addition to more than one way of representation of some of the interrogatives. For instance, interrogatives like Who "من", Whose "لمن", What "ما" "ماذا" "مما", Which "أي", Where "أين", When"متى" "أيّان", Why "لماذا", How (much/many) "كم" and How "كيف", are some ways of asking questions in Arabic. In addition, Arabic Interrogative nouns include other particles that are related to the expected question types and/or scopes, like:

1) To be type "هل": This is an interrogative tool where such type of question usually would be looking for a Yes/No answer. Examples on such: "هل تمطر في الخارج؟" (Is it raining outside?) and "هل اتى محمد؟" (Did Mohammad come yet?).
2) Hamza "ء": This interrogative tool in Arabic is used as a disapprovingly tool "أداة استنكارية". As an example: in the Holy Quran "ءإله مع الله؟" (Is there a God with Allah?), the answer to such questions would be either Yes or No, but in our example the answer must definitely be No.

The Arabic interrogative word "مما" is actually a combination of the two words "من" and "ماذا" (From What), but when combined in Arabic it is reduced to "مما".

In addition to the above, Arabic has more indirect ways to ask questions. For instance, in Arabic, using the phrase "في أي" (In What/Which), we could say "في أي عام ولد ابن خلدون؟" In What year Ibn Khaldoun was born? Or "في أي بلد ولد ابن خلدون؟" (In which country was Ibn Khaldoun born?). Such questions are formulated using the phrase: "في أي" (In What/Which); a phrase used to formulate question types related to Time (In What) and Place (In Which) concepts. Also, one can use the phrase "من أين" (From Where) to usually refer to a source (Location, Method, …etc.) like asking"من اين اكتسبت هذا المال؟" (From where did you earn this money?); referring to the source of the money, or ask "من أين أتى الغزاة؟" (From where did the attackers come?); referring to the location the attackers came from.

On the other hand, the phrase"فيـمَ" (In Where) refers to questions that ask about a target Named Entity like "فيما انفقت مالك؟" (Where did you spend your money?). This type of question might require some extra semantic analysis and search capabilities to reach a proper answer.

In Arabic, once the type of a question is identified, from the interrogative noun we can identify the Gender and the expected scope (Target Answer type); which would be used later in the Response Generation module; another part of the ongoing research on Arabic QA. This can be illustrated in formulating what is known as Question Patterns. For instance, if one asks the following question:

"من هو محمد الفاتح؟"(Who is Mohammad Al-Fatih?)

then using the following Question Pattern for Definitional type of question Who+be +< topic> (من هو|هي + <الموضوع>), the question type and scope would be identified from the usage of the interrogative noun "من" as being a Definitional type of question looking for an NE→ Enamex. From the word "هو", referring to the question pattern would indicate that gender is identified as masculine. The Target Answer (topic) would be referring to the Named Entity "محمد الفاتح" Mohammad Al-Fatih. Furthermore, a question like:

"متى حدثت معركة الكرامة؟" (When did Al-Karamah Battle occur?)

then using the following Question Pattern for Temporal type of question: When+Verb+<topic> (متى+ فعل + <الموضوع>), the question type and scope would be identified from the usage of the pattern of "متى" as being a Factoid type looking for an NE → Timex and the gender would be identified as feminine from both the Taa marboutah "ـة" in "معركة الكرامة" and the connected pronoun "ت" in the verb "حدثت". The Target Answer (topic) will be looking for a Time or Date value related to the occurrence of the scope "معركة الكرامة"; i.e., a year or a specific date. Table 3 gives a sample of Question Patterns that were identified and used for this purpose.

Table 3. Sample of Question Patterns used in the approach.

Question Pattern	Type	Scope
من + هو | هي + <الموضوع>	Definitional	Definition of NE→ Enamex
لمن + إسم اشارة + <الموضوع>	Factoid	NE→ Enamex
لمن + <الموضوع> + اسم إشارة	Factoid	NE→ Enamex
ما + هو|هي + <الموضوع>	Definitional	Definition of NE
كم [تبلغ | يبلغ] + Countable-Qty.-Term + < تكملة السؤال >	Factoid	NE→ Numex
متى + فعل + <الموضوع>	Factoid	NE→ Timex
كيف + فعل + <بقية السؤال>	Method	List of steps

3.1.1 Question Processing

To process a question, a set of rules and patterns were developed for each question type. Table 4 gives Question types and scope. The current version of the implemented system can answer some of the rules for (How and Why) "كيف، لماذا". Further assessment to produce more accurate answers is needed; which constituted part of the future research as well. The rules for the interrogative noun (List) "اذكر" were not tested in our approach, since they require more semantic analysis. Different rules for interrogative nouns in Arabic were built and implemented within the system. Figure 1 and Figure 2 give the rules used for "من" and "متى، إيان، أيان", respectively.

So far, the rule in Figure 1 deals with the question pattern (من هو|هي + <الموضوع>), in which the second token of the given question must be either "هو" or "هي". The rule for the case where the interrogative noun "من" followed by a verb was not built and implemented, since it requires more analysis. So, if a question like من جاء مع محمد إلى الجامعة ؟ (who came with Mohammed to the University) was asked, then it will not be answered, as it is not handled properly in the current approach due to more analysis. Figure 3 describes the question processing applied in this approach.

Table 4. Question types and scope in Question Understanding.

Question Type	Question Words	Scope
Factoid	ايان ، إيان ، متى	Timex, looking for a time
Factoid	لمن	Enamex, looking for Named Entity → Person
Factoid	في اي + Time_Term	Timex, looking for a time
Factoid	في اي + Location_Term	Enamex, looking for a Location
Factoid	اين	Enamex, looking for a Location
Factoid	كم	Looking for Numeric Value
Definition	من	Enamex, looking for definition of Named Entity → Person
Definition	ما	Enamex, looking for definition of Named Entity → Location or Organization
Causal	ماذا	Looking for result and/or cause
Method	كيف	Looking for a process to do something
Purpose	لماذا	Looking for a reason for doing something
List	اذكر، ما هي	Looking for a list of steps. In case ما هي, we need further analysis like next word is طريقة، مقادير، خطوات
Yes/No	هل	Looking for either Yes or No as a reflection of action

```
If Question First Token = "من" then
  If Second_Token = "هو" or Second_Token = "هي" Then
        Question_Type = Definition;
         Question_Scope = Enamex_Person
        Question_Scope_Word = Named_Entity(ies) after Second_Token
        Expected_Answer = Definition of the Named Entity of Person
        Extract and Build Question Keyword List
  End if
End if
```

Figure 1. Rule for interrogative noun "من".

```
If Question First Token = "متى" or "أيان" or "ايان" then
        Question_Type = Factoid;
        Question_Scope = Timex_Time_or_Date
        Main_Verb = Second_Token
        Question_Scope_Word = Numeric Value of Time related to
              Main_Verb occurrence
        Expected_Answer = Number or Sentence containing Time term
        Extract and Build Question Keyword List
        Enrich Question Keywords with Time_Related Terms
End if
```

Figure 2. Rule for interrogative nouns "متى", "إيان" or "أيان".

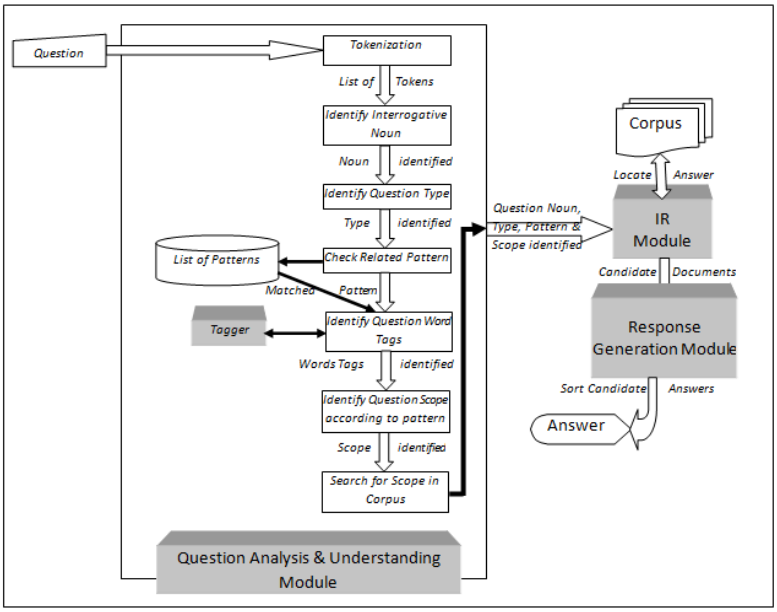

Figure 3. Question processing of the research.

3.1.2 The Tagger System

Arabic is a highly inflectional and derivational language, which exists in three different forms today. The first is the Classical Arabic (Fus'ha), which is the language of the Holy Quran and poetry of ancient pre-Islam era before about 1500 years. This form is completely diacriticized and is used today in the teachings of the Holy Quran and any Islam-related issues as well as Arabic Language educational books. The second form is the Modern Standard Arabic (MSA), which is the language of the media (Newspapers, TV, Internet, …etc.) and is used daily for official purposes by

Arab countries and their organizations as well as the United Nations as one of the official languages. Finally, the third form is the Colloquial Language (CL), which refers to different dialects that are used in different Arab countries amongst people on the streets [38].

The majority of research in Arabic NLP in existence today has targeted Modern Standard Arabic; a vowel free form of the language that introduces high ambiguity. For instance, if we look at the MSA word "علم", without diacritic symbols shown on the word, it could be interpreted to mean 'Science' "عِلْم" (Ilm), 'Flag' " عَلَمْ" (Alam), 'been known' (عُلِمْ) (Olim), …etc. An example of similar derivations is given in Table 5 for the root ق ب ل [39]. This ambiguity is one of the main reasons why research on Arabic NLP did not reach the levels of its counterpart Latin-based languages.

Table 5. Different derivations of the root ق ب ل [39].

English Concept	Arabic word
Tribe	قبيلة
to meet	تقابل
Before	قبل
Future	مستقبل
To receive	استقبل
To come to	أقبل
Kiblah	قبلة

Arabic words are either native Arabic words or Arabized; brought from other languages. According to [40], the following 12 letters (ض ط ظ ق ع ح ة ء ؤ ئ ى ص) are restricted to Arabic native words where none of them is used for either Transliterated and/or Arabized words. An Arabic native word is mainly classified into one of three types; Noun, Verb and Particle. When performing NLP using Modern Standard Arabic form; especially in Question Answering systems, it is very important to identify the type of the word we are dealing with; especially nouns. Figure 4 gives the Part-of-Speech categorization of Arabic words [41].

Of course, before we could identify Target Answer types, we need to tag question words properly; and this is the task of the Tagger. In this research, we have built a tagger that is a combination of both rule-based and word weights "أوزان" to identify the proper tag value of a question word. The used rules and weights, and hence the built tagger, are from two unpublished research works by the author.

In this research, 35+ tagging rules were identified from literature that could be used for the proper identification of Nouns. For instance, the following types of words are all identified as Nouns: Proper nouns, Action nouns, Genus nouns, Agent nouns, Patient nouns, Adjectives, Time, Place, Instrument, Adverbs and Demonstrative Nouns. In addition, any words that start with the definite article "ال" (the) or end with "ـة" are considered Nouns. In most cases, words that end with "ـاء" are considered nouns. In this research, we used question patterns which enforce us to use nouns and not verbs that end with "ـاء". If tagging rules failed in identifying Nouns properly, then a set of Named Entities that was compiled by [31] are used to help in properly classifying Named Entities. In addition, 15+ rules are identified to tag verbs and 12+ rules are identified and used to identify particles. Such tagging rules are needed to help in proper identification of answers. In addition to the rules, 72+ word weights were used to help in the proper identification of word tags and are implemented within the system.

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 2, No. 3, December 2016.

The tagging algorithm is given as follows:

1) Get the word from the Tokenized list of words.
2) Assign the value “Unknown” to Tag_value .
3) Check if word is a Particle. If so, assign Tag_value = “Particle” and go to step 11.
4) Check if word is a Noun. If so, assign Tag_value = “Noun” and go to step 7.
5) Check if word is a Verb. If so, assign Tag_value = “Verb” and go to step 11.
6) If Tag_value equals “Unknown”, Check the NE database for a match.
7) If the word is found in the Location NE table, assign Tag_value = “NE_Loc” and go to step 11.
8) If the word is found in the Person NE table, assign Tag_value = “NE_Hum” and go to step 11.
9) If the word is found in the Organization NE table, assign Tag_value = “NE_Org” and go to step 11.
10) If Tag_value is still “Unknown”, assign “Failed or Foreign” to Tag_value.
11) Return Tag_value and Exit.

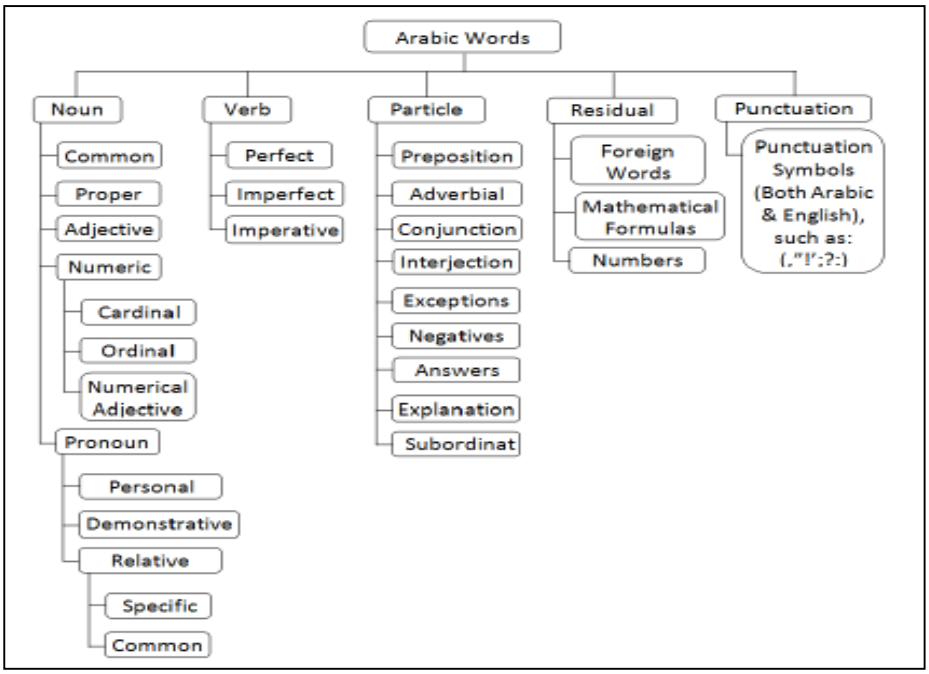

Figure 4. Arabic Part-of-Speech [41].

In this algorithm, the Tag_value is needed for proper identification of question words to better match question types and scope with Question Patterns. If the Tag_value was identified as either Particle or Verb, then the algorithm just returns the Tag_value. If, however, the Noun was identified as a Tag_value, or the Tag_value was not identified (Unknown Tag_value), then a search is needed

into the Named Entity tables to better identify the Question Scope; especially with Factoid and Definitional types of questions. Figure 5 gives a snapshot of Noun tagging rules.

```
If word starts with a letter in "سألتمونيها" then
  If first_letter in "س،ل،م،و،ه" then the word is noun,
    return Tag_value="Noun"
  Else if first_letter in "ا،أ،ن،ت،ي" then
        If second_letter is not in "سألتمونيها", then
            Assume word is Verb and go to Third_Test
        Else if second_letter is not equal "س" then the word is noun,
                return Tag_value="Noun"
            Else if third_letter = "ت" then
                Assume word is Verb and go to Third_Test
```

Figure 5. Snapshot of Noun tagging rules.

3.2 The Information Retrieval/Extraction Module

The focus of this research did not involve building a state-of-art IR engine. For this, we have adopted the approach followed by many researchers like that of QARAB, QArabPro, …etc. that was based on Salton Vector Space Model (VSM) to search and retrieve relevant documents using a relational database system. In this system, we keep data in tables where the major tables are:

1) A Document table, where we store different corpora files into the database.
2) A Stop words table to keep a list of Arabic Stop words. Here, we use a list of Arabic Stop words that was downloaded from [42] and contains 13000+ Stop words.
3) A Named Entity table, which stores a list of Named Entities. Here, we use the list compiled by [31] that contains 5000+ person, location and organization names.

In addition to the above mentioned tables, the approach includes some other supporting tables that are needed during the Question Analysis and Information Retrieval process.

The Vector Space Model (VSM) defines a vector that represents each document and a vector that represents the query. The model works by assigning weights to index terms in both the queries and the documents, which are then used to calculate the degree of similarity between each document and the query. In this research, the Cosine similarity is used as a measure of similarity between the query and the retrieved relevant documents to find the most appropriate answers.

After the proper formulation of the query posed by the question through enriching it with extra keywords, the IR model retrieves the most relative documents to the query that need to further choose among. The choice is made after sorting documents using some scoring mechanism. Given a document dj and a query q, the Cosine Similarity value that could be used to provide such score is calculated as:

$$CosSim\left(d_j, q\right) = \frac{d_j . q}{\left|\overrightarrow{d_j}\right| . \left|\overrightarrow{q}\right|} = \frac{\sum_{i=1}^{t}\left(W_{ij} . W_{iq}\right)}{\sqrt{\sum_{i=1}^{t} W_{ij}^2 . \sum_{i=1}^{t} W_{iq}^2}};$$

where w_{ij} is the weight of the term i in the document j and w_{iq} is the weight of the term i in the query. The term weight is the Normalized term weight and is calculated as:

$f_{i,j} = tf_{i,j} / \max tf_{i,j}$, where:

$f_{i,j}$ =Normalized frequency,

$tf_{i,j}$ = Frequency of term i in document j and

$\max tf_{i,j}$ = Maximum frequency of term i in document j.

Since we will be looking for the most appropriate answer, then the document with the highest score will be selected as the candidate document containing the answer. Once the document is identified, a pattern matching is performed to find the best match between the different sentences in the document and the pattern scope and shape to retrieve the best answer.

4. THE DATA SET

For the purposes of this research, we have started looking for different sources of Data Sets that could be used. Although many corpora were collected, only the Data Set built for QArabPro by [14] was used, due to an access that was thankfully obtained to both the questions and the documents from the lead author. The Data Set consists of 335 questions posed over 74 documents from different categories; which has formulated the basis for testing and evaluating our approach. In addition, ANERSys Named Entity corpus built by [31] was used to help in the tagging process. The testing results are reported in the experimental section of this paper.

After extensive search of the Internet and repositories, other corpora were also collected like that of the Holy Quran [43], List of Stop words from [42] and a corpus of 1256 documents collected by the author from both Al-Rai (www.alrai.com) and Addustour (www.addustour.com) newspaper sites. However, such corpora were not used in this research due to the lack of experimental results to compare with and to make sure that the approach of this research really works. As a future research, the author is planning to use such corpora and to apply the approach upon them.

5. RESEARCH APPROACH

To achieve the objectives of this research, the following steps were followed as given in Figure 6.

Step 1) Collecting and organizing information on different Arabic Question Answering systems in existence with their related problems.

After extensive search of existing Arabic QA systems, the main problems found to be faced by such Arabic QA systems could be summarized as follows:

(1) The lack of standardized Arabic resources, like an Arabic corpora, Grammar, IR tools, …etc., that could be used as a bench mark to compare with and judge the effectiveness of an approach.

(2) Some of the discussed QA systems did not report their testing results or even did not mention anything about their results, like AQAS and QARAB.

(3) The majority of systems are of ***Factoid*** and/or ***Definitional*** type of QA systems. Very few of them managed to handle the "Why and How" type of question like QArabPro and none were found to deal with ***List*** type of question. This is because more processing and semantic analysis is requested, which requires more elaborative work.

(4) The majority of the systems assume that the answer exists in the set of documents being searched; a limitation of the QA system. But, what if the answer does not exist? None of the researchers said anything about that.

Step 2) Identifying different rules in existence for Analyzing Arabic Questions.

To perform this process, the literature was searched for existing rules that could be helpful and useful for the purposes of Analyzing Arabic Questions. Most of the rules had to deal with the tagging process of Arabic words and a very limited number of rules was clearly mentioned in the papers (5 rules only by [14]). Some authors did not even mention the way in which they have analyzed Arabic questions in their systems. For this, we had to build our own set of rules that matches the logic of dealing with different forms of Questions.

Step 3) Enhancing and/or building rules for Question Analysis.

By analyzing different ways of asking questions in Arabic and from different sources in literature, we have identified six different categories of question types; those are: Factoid, Definitional, Causal, Method, Purpose and List. Although in English type of questions there are six different Question words; Who, When, Where, Which, Why and How, their Arabic counterparts constitute around 15 different variations. The Arabic question words are: من، متى، أين، أي، لماذا، كيف، ممَّ، في اي، ماذا، هل، اذكر، أيان، كم with two variations for each of في أي and كم. Rules were built for each type of Arabic questions.

Step 4) Building and/or collecting proper corpora for testing purposes.

Many corpora were collected from different resources of the Internet and from different scholar sites. The total number of documents in these corpora amounted to more than 50000 text documents in addition to the Holy Quran in Arabic textual format. However, to test the validity of our approach, only the corpus provided by [14] is used.

Step 5) Building an Arabic QA system equipped with the new rules to test the approach.

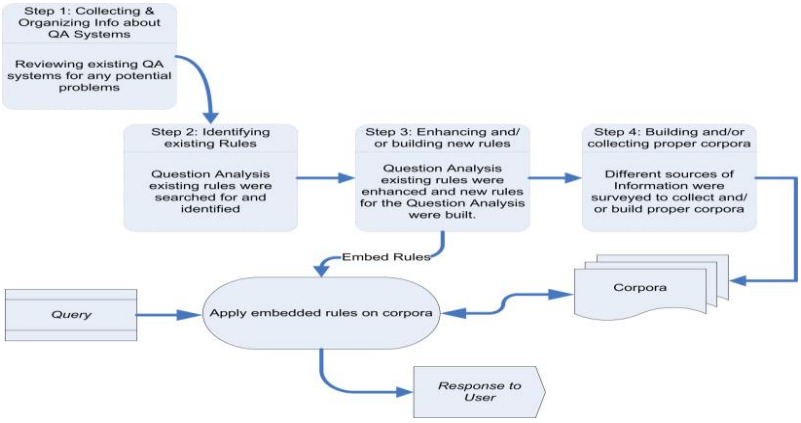

Figure 6. Generic steps of research approach.

6. IMPLEMENTATION

In this research, a rule-based QA system using Visual Studio.Net 2015 with SQL Server 2014 Express Edition within a DotNet framework version 4.6 under Windows 7 environment was implemented for testing purposes. The system was implemented using the identified and constructed rules. The main screen of the system is given in Figure 7. From this figure, we can notice that the user has the choice to either ask a question directly; by selecting Question option, or load a set of questions from a question file; by selecting the Question File option. A user must identify the corpus to be used for the search purposes. The main reason behind selecting the proper

corpus is to work in the same manner as that of the QArabPro for bench marking purposes. Furthermore, to select a file containing a set of questions was a choice to compare the results with those of QArabPro.

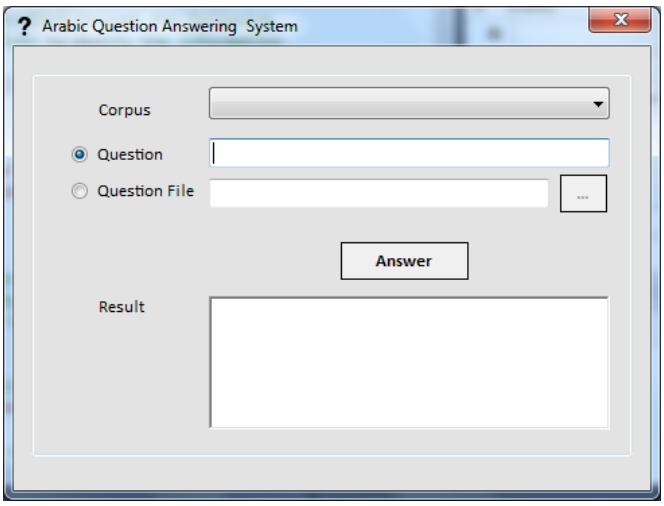

Figure 7. Main screen of the implemented system.

The system makes use of the documents and the Named Entities stored into the database as of well as other tables mentioned previously in the section about the IR module. No document pre-processing or letter normalization is performed. Instead, when storing the document into the database, we store a document ID, a category for which the document belongs and the document text as is.

When conducting research on Arabic NLP, many researchers tend to perform normalization of some letters by converting different versions of alif "ا أ آ إ" into a single version "ا" as well as the case for haa "ه ـة" and alif maqsourah "ى ي". In the author's opinion, such process will lead to invalid answers in a question answering system and Figure 8 gives the proof. This opinion matches with a finding by [44], in which is proved that the removal of Stop words and Normalization of letters had no significant effect on the retrieval process and might not justify the cost of carrying pre-processing.

In Figure 8, if one asks about Arwad island using alif with hamzah "ما هي جزيرة أرواد؟" and alif without hamzah "ما هي جزيرة ارواد؟", the results will be completely different; with the one containing the hamzah as the correct answer. If the corpus was written properly without typos, then the question without hamzah will not obtain any answer; since Arwad is usually written with a hamzah.

If normalization was performed on both the documents and queries, incorrect answers might be obtained (as in Figure 8). By eliminating the normalization process and using the question patterns, better results are obtained.

The IR module of the approach was implemented based on the Vector Space Module (VSM) to search for answers among the set of documents and then rank them using the Cosine Similarity measure, in which the document with the highest earned value is selected as a candidate document. To extract the answer, the system performs pattern matching between the question pattern and

different sentences in the document that contain the question scope. The sentence that contains the scope and best matches the pattern is returned as the answer.

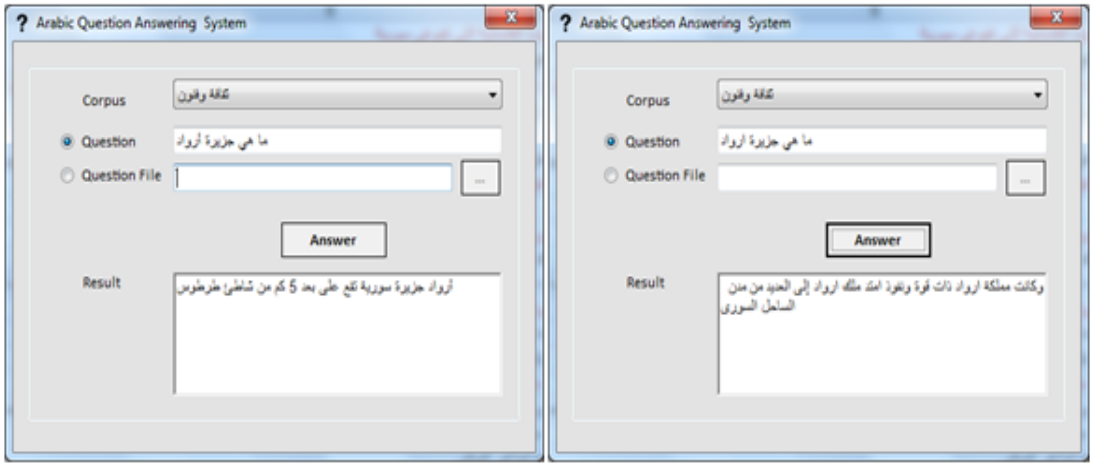

Figure 8. Reason for not normalizing Arabic characters.

7. EXPERIMENTAL RESULTS

To test this approach, the data set used for QArabPro in [14] was used as a bench mark. In this data set, a set of 335 questions and 74 documents were used. However, before we could use this data set, we had to convert all text documents from Windows 1256 code page into UTF-8 encoding. This was a necessity for the success of dealing with this corpus under Windows 7 environment. In addition, the set of documents were stored into the database in their original format where a document ID, category and text were stored. Table 6 gives more information on the number of questions of each question type adopted in this approach from [14].

With IR systems, both Precision (P) and Recall (R) are used as measures to show the efficiency of such systems in retrieving relative documents. When calculating both P and R, it is clearly noted that both have an inverse relationship; if P increases, R should decrease. However, since Question Answering is usually interested in finding an exact answer; not a document (*or list of documents*), then using P; in the author's opinion, will not be an accurate measure, as it will be difficult with such case to identify True Negative and False Positive answers needed to calculate the Precision. When posing a question, only one of three outcomes will be noticed: a correct answer, an incorrect answer or no answer. As is the case with many researches on QA, we used Accuracy (Acc) instead of Precision; which refers here

to the ratio of correctly answered questions over the total number of questions posed, as a measure. The following formulae show how we measured the efficiency of the approach.

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 2, No. 3, December 2016.

Table 6. Question distribution per question type.

Question Type	Total # Questions	Total # Answered	Correctly Answered	Incorrectly Answered	Not Answered	Accuracy
Definitional	141	136	118	23	5	0.867647
Factoid	128	125	86	36	3	0.688
Causal	40	39	32	6	1	0.820513
Purpose	26	25	18	6	1	0.72
Total	335	325	254	71	10	**0.781538**

$$Recall = \frac{\text{Number of answered questions}}{\text{Total number of asked questions}} = \frac{325}{335} = 97\%$$

$$Accuracy = \frac{\text{Number of correctly answered questions}}{\text{Total number of answered questions}}$$

$$Definitional\ Accuracy = \frac{118}{136}$$

$$Definitional\ Accuracy = 0.867647$$

$$Factoid\ Accuracy = \frac{86}{125}$$

$$Factoid\ Accuracy = 0.688$$

$$Causal\ Accuracy = \frac{32}{39}$$

$$Causal\ Accuracy = 0.820513$$

$$Purpose\ Accuracy = \frac{18}{25}$$

$$Purpose\ Accuracy = 0.72$$

$$Overall\ Accuracy = \frac{254}{325}$$

$$Overall\ Accuracy = 0.781538$$

$$Fmeasure = 2 * \frac{\text{Accuracy} * \text{Recall}}{\text{Accuracy} + \text{Recall}} = 2 * \frac{0.781538 * 0.970149}{0.781538 + 0.970149}$$

$$Fmeasure = 0.8656869642.$$

To reach the obtained results, we have isolated different questions per each category into files and then used these files as input to the system. The results were stored in an Excel file showing each question with its corresponding answer. A manual process was then performed by a human expert to validate the answers given by the system. Finally, all results of categories were combined into one file. The number of correctly answered questions, as well as the number of incorrectly answered questions and the number of unanswered questions were manually calculated. Figure 9 shows a snapshot of the combined results in which answers in light colour are incorrect, empty cells indicate unanswered questions, while answers in dark colour refer to correctly answered questions.

Figure 10 shows the number of questions that were answered from each type of question posed and Figure 11 gives the percentage of correctly answered questions for each question type. As can be

noticed from Figure 11, the system managed to answer around 72% of Purpose type of questions, 82% of Causal type of question, 87% of Definition type of question and 68.8% of Factoid type of question. Only 2.9% of questions were not answered and 21.2% were incorrectly answered.

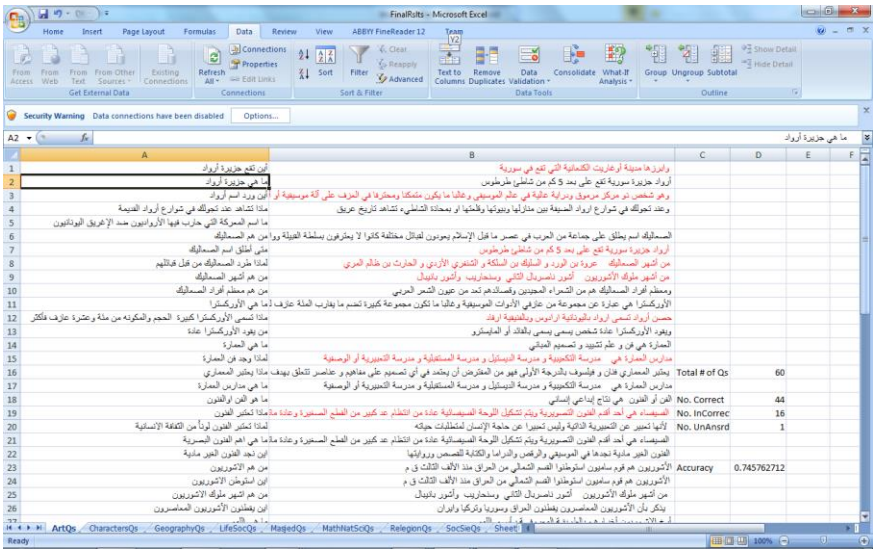

Figure 9. Snapshot of the combined results.

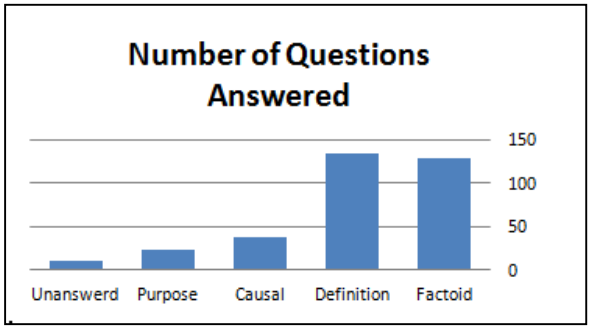

Figure 10. Number of questions answered by the QA system.

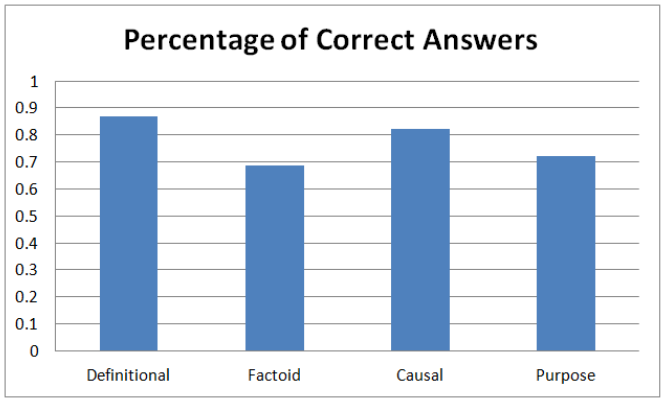

Figure 11. Percentages of correct answers among different categories.

8. SUMMARY AND CONCLUSION

In this research, a set of rules for the Analysis and Understanding of Arabic Questions in an Arabic Question Answering environment was built. To achieve the purpose of this research, different tagging rules as well as question patterns that could help in locating a more accurate answer were also built.

In comparison to the work of [14] which was used for benchmarking purposes, our approach has obtained better recall (97% vs 86%). In addition, our approach has managed to obtain better accuracy for some types of questions. For instance, our approach has obtained an accuracy of 75% for "كم" type and 72% for "لماذا" type of questions in comparison to 69% and 62% respectively in [14]. Accuracy of other types of questions was not mentioned in [14], so we could not compare our results with theirs. In our opinion, lower overall accuracy of 78% obtained by our approach, as well as low accuracy results obtained for some type of questions; especially for Factoid types (86 out of 125), can be referred to the content of the data set where many typos were found in both the text documents and the formulation of the questions. So, to obtain better results, the documents and questions need to be revised.

The scope of this part of research has concentrated on the Question Analysis and Understanding module. The IR module, however, was built using the approach used by other authors; i.e., the Relational Database approach. The Cosine similarity over the Salton VSM module was used to rank different candidate answer documents. Testing results showed an overall accuracy of 78% with a recall of 97% and an F-Measure of about 87%.

9. FUTURE RESEARCH

It can be noted that not all rules were fully constructed and implemented in this approach. For instance, rules matching a pattern like (من + فعل + <الموضوع>) are not handled in this approach. For this, we are planning to expand and extensively review and enhance all built rules to obtain better answers and performance. We are also planning to prepare and double check the collected corpora for any typos to be used for testing purposes and to make sure that our approach is performing well by manually double checking the expected answers from each of the questions. This step will be used toward applying and generalizing our approach on other collected corpora.

With some variations to the approach and its rules, work on obtaining answers from the Holy Quran [43], would constitute another part of future research. In addition, we are planning on extending the approach to deal with the sayings of the Prophet Mohammad (Peace be Upon Him).

The current research is based on syntactic analysis of words. As part of future research, we are planning on using the semantic analysis as well to help in locating proper answers.

Since this research constitutes part of an ongoing research, we are currently working on building proper response generation rules that can be used to give better answers in a dialog like context with the user.

REFERENCES

[1] A. Ezzeldin and M. Shaheen, "A Survey of Arabic Question Answering: Challenges, Tasks, Approaches, Tools and Future Trends," Proc.13th International Arab Conference on Information Technology (ACIT 2012), Paper ID 13106, Zarqa University, Jordan.

[2] C. L. Paris, "Towards More Graceful Interaction: A Survey of Question-Answering Programs," Technical Report, Columbia University, Report no. CUCS-209-85, 1985.

[3] M. R. Kangavari, S. Ghandchi and M. Golpour, "Information Retrieval: Improving Question Answering Systems by Query Reformulation and Answer Validation," Journal of World Academy of Science: Engineering & Technology, pp. 303-310, 2008.

[4] N. Kuchmann-Beauger and M. A. Aufaure, "A Natural Language Interface for Data Warehouse Question Answering," Natural Language Processing and Information Systems, pp. 201-208, 2011.

[5] D. Tufiş, "Natural Language Question Answering in Open Domains," Computer Science Journal of Moldova, vol. 19, no. 2, 2011.

[6] S. Mittal and A. Mittal, "Versatile Question Answering Systems: Seeing in Synthesis," International Journal of Intelligent Information and Database Systems, vol. 5, no. 2, pp. 119-142, 2011.

[7] S. Blair-Goldensohn, K. R. McKeown and A. H. Schlaikjer, "Defscriber: A Hybrid System for Definitional QA," Proc. 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 462-462, 2003.

[8] M. R. Kangavari, S. Ghandchi and M. Golpour, "A New Model for Question Answering Systems," Journal of World Academy of Science: Engineering & Technology, vol. 2, no. 6, pp. 506-513, 2008.

[9] A. Monroy, H. Calvo and A. Gelbukh, "NLP for Shallow Question Answering of Legal Documents Using Graphs," Computational Linguistics and Intelligent Text Processing, pp. 498-508, 2009.

[10] C. Unger and P. Cimiano, "Pythia: Compositional Meaning Construction for Ontology-based Question Answering on the Semantic Web," Natural Language Processing and Information Systems, pp. 153-160, 2011.

[11] F. A. Mohammed, K. Nasse and H. M. Harb, "A Knowledge Based Arabic Question Answering System (AQAS)," ACM SIGART Bulletin, vol. 4, no.4, pp. 21-30, 1993.

[12] B. Hammo, H. Abu-Salem and S. Lytinen, "QARAB: A Question Answering System to Support the Arabic Language," Proc. ACL-02 Workshop on Computational Approaches to Semitic Languages, pp. 1-11, 2002.

[13] Y. Benajiba, P. Rosso and A. Lyhyaoui, "Implementation of the ArabiQA Question Answering System's Components," Proc. Workshop on Arabic Natural Language Processing, 2nd Information Communication Technologies Int. Symposium, ICTIS-2007, Fez, Morroco, 3-5 April 2007.

[14] M. Akour, S. Abufardeh, K. Magel and Q.Al-Radaideh, "QArabPro: A Rule Based Question Answering System for Reading Comprehension Tests in Arabic," American Journal of Applied Sciences, vol. 8, no. 6, pp. 652-661, 2011.

[15] S. Bekhti, A. Rahman, M. Al-Harbi and T. Saba, "AQUASYS: An Arabic Question-Answering System Based on Extensive Question Analysis and Answer Relevance Scoring," International Journal of Academic Research, vol. 3, pp. 45-54, 2011.

[16] O. Trigui, L. H. Belguith, P. Rosso, H. B. Amor and B. Gafsaoui, "Arabic QA4MRE at CLEF 2012: Arabic Question Answering for Machine Reading Evaluation," Proc. CLEF (Online Working Notes/ Labs/ Workshop), 2012.

[17] N. Fareed, H. Mousa and A. Elsisi, "Enhanced Semantic Arabic Question Answering System Based on Khoja Stemmer and AWN," Proc. 9th International Conference on Computer Engineering, (ICENCO-2013), pp. 85-91, 2013.

[18] H. Kurdi, S. Alkhaider and N. Alfaifi, "Development and Evaluation of a Web Based Question Answering System for Arabic Language," International Journal on Natural Language Computing (IJNLC), vol. 3, no. 2, 2014.

[19] V. Guda, S. Sanampudi and L. Manikyamba, "Approaches for Question Answering," International Journal of Engineering Science and Technology (IJEST), vol. 3, no. 2, 2011.

[20] W. Bdour and N. Gharaibeh, "Development of Yes/No Arabic Question Answering System," International Journal of Artificial Intelligence & Applications (IJAIA), vol. 4, no. 1, 2013.

[21] O. Al-Harbi, S. Jusoh and N. Norwaw, "Handling Ambiguity Problems of Natural Language Interfaces for Question Answering," International Journal of Computer Science Issues, vol. 9, no. 3, pp. 17-25, 2012.

[22] A. Azmi and N. Al-Shenaifi, "Handling 'Why' Questions in Arabic," Proc. 5th International Conference on Arabic Language Processing, (CITALA 2014), pp. 206-209, 2014.

[23] P. Rosso, Y. Benajiba and A. Lyhyaoui, "Towards an Arabic Question Answering System," Proc. 4th Conference on Scientific Research Outlook & Technology Development in the Arab World (SROIV), Damascus, Syria, 2006.

[24] Y. Benajiba, P. Rosso and J. M. Benedíruiz, "ANERsys: An Arabic Named Entity Recognition System Based on Maximum Entropy," Computational Linguistics and Intelligent Text Processing, pp. 143-153, 2007.

[25] W. Brini, M. Ellouze, S. Mesfar and L. H. Belguith, "An Arabic Question-Answering System for Factoid Questions," Proc. International Conference on Natural Language Processing and Knowledge Engineering (NLP-KE), pp. 1-7, 2009.

[26] G. Kanaan, A. Hammouri, R. Al-Shalabi and M. Swalha, "A New Question Answering System for the Arabic Language," American Journal of Applied Sciences, vol. 6, no. 4, pp. 797-805, 2009.

[27] O. Trigui, L. H. Belguith and P. Rosso, "DefArabicQA: Arabic Definition Question Answering System," Proc. 7th Workshop on Language Resources and Human Language Technologies for Semitic Languages (LREC), Valletta, Malta, pp. 40-45, 2010.

[28] L. Abouenour, K. Bouzoubaa and P. Rosso, "IDRAAQ: New Arabic Question Answering System Based on Query Expansion and Passage Retrieval," Proc. CLEF 2012 Workshop on Question Answering for Machine Reading Evaluation (QA4MRE), 2012.

[29] A. Ezzeldin, M. Kholief and Y. El-Sonbaty, "ALQASIM: Arabic Language Question Answer Selection in Machines, Information Access Evaluation, Multilinguality, Multimodality and Visualization," Lecture Notes in Computer Science, vol. 8138, pp. 100-103, 2013.

[30] F. Al-Khawaldeh, "Answer Extraction for Why Arabic Question Answering Systems: EWAQ," Proc. World of Computer Science and Information Technology Journal (WCSIT), vol. 5, no. 5, pp. 82-86, 2015.

[31] K. Shaalan and H. Raza, "NERA: Named Entity Recognition for Arabic," Journal of the American Society for Information Science and Technology, vol. 60, no. 8, pp. 1652-1663, 2009.

[32] J. Maloney and M. Niv, "TAGARAB: A Fast, Accurate Arabic Name Recognizer Using High-precision Morphological Analysis," Proc. Workshop on Computational Approaches to Semitic Languages, pp. 8-15, 1998.

[33] K. Shaalan and H. Raza, "Person Name Entity Recognition for Arabic," Proc. 2007 Workshop on Computational Approaches to Semitic Languages: Common Issues and Resources, pp. 17-24, 2007.

[34] C. Shihadeh and G. Neumann, "ARNE: A Tool for Named Entity Recognition from Arabic Text," Proc. 4th Workshop on Computational Approaches to Arabic Script-based Languages (CAASL4), Located at the 10th Biennial Conference of the Association for Machine Translation in the Americas (AMTA), pp. 24-31, 2012.

[35] Y. Benajiba, M. Diab and P. Rosso, "Using Language Independent and Language Specific Features to Enhance Arabic Named Entity Recognition," Proc. International Arab Journal of Information Technology (IAJIT), vol. 6, no. 5, 2009.

[36] Y. Benajiba, I. Zitouni, M. Diab and P. Rosso, "Arabic Named Entity Recognition: Using Features Extracted from Noisy Data," Proc. ACL 2010 Conference Short Papers, ACLShort, Stroudsburg, PA., pp. 281–285, 2010.

[37] Y. Benajiba and P. Rosso, Arabic Question Answering, Diploma of Advanced Studies, Technical University of Valencia, Spain, 2007.

[38] K. Shaalan, "A Survey of Arabic Named Entity Recognition and Classification," Proc. Computational Linguistics, vol. 40, no. 2, 2014.

[39] Microsoft Arabic Word-Breaker, white paper, http://www.microsoft.com/en-ph/download/confirmation.aspx?id=32828, (accessed June 23rd, 2015).

[40] M. Attia, A. Toral, L. Tounsi, M. Monachini and J. Van Genabith, "An Automatically Built Named Entity Lexicon for Arabic," Proc. 7th Conference on International Language Resources and Evaluation (LREC'10), Valletta, Malta, European Language Resources Association (ELRA), May 2010.

[41] I. Gharaibeh and N. Gharaibeh, "Towards Arabic Noun Phrase Extractor (ANPE) Using Information Retrieval Techniques," Software Engineering, vol. 2, no. 2, pp. 36-42, 2012.

[42] Tenzil.net web site, Quran Corpus from http://tanzil.net/download/ (accessed on June 16th, 2015).

[43] Arabic Stop words https://sourceforge.net/projects/arabicStop words/ (accessed on July 26th, 2015).

[44] M. Al-Nabhan, An Investigation of the Impact of Stop Words Removal and Word Normalization on the Performance of Stem-based Arabic Information Retrieval, Unpublished MSc Thesis, Computer Information Systems Department, Faculty of IT, Yarmouk University, December 2015.

ملخص البحث:

يواجه البحث في معالجة اللغة العربية العديد من المشاكل التي تسببها صعوبة اللغة، وقلة الموارد المقروءة آلياً، وقلة الاهتمام من الباحثين العرب. يعد حقل السؤال والجواب أحد الحقول التي بدأت عمليات البحث الظهور فيه. وعلى الرغم من وجود بعض البحوث في هذا المجال، فإن القليل منها فقط أثبتت فعاليتها في إيجاد الإجابة الصحيحة. وتعد عمليات تحديد نوع العناصر وتحليل السؤال وفهمه من الأمور التي أدت إلى التأثير في دقة الإجابات المستخرجة. تم في هذا البحث بناء مجموعة من 60+ من القواعد المناسبة لتحديد نوع العناصر ومجموعة من 15+ من القواعد لتحليل الأسئلة بطريقة صحيحة، و20+ من قوالب الأسئلة لتحسين عملية الوصول إلى النتائج المطلوبة بشكل صحيح من وثائق معلومات تم جمعها من مصادر مختلفة. ولإثبات فعالية القواعد، تم بناء نظام للأسئلة والأجوبة وتم الوصول إلى دقة إجمالية بنسبة تبلغ حوالي 78% واسترجاع بنسبة 97% و مقياس (F) بنسبة تقرب من 87%.