Documentation

[An Arabic Question-Answering system]

Team

1. Adham Mohamed (sec 2)
2. Ehab Ahmed (sec 4)
3. Ibrahim Mohamed (sec 1)
4. Aya El-Sayed (sec 4)
5. Amira Ramadan (sec 4)

Index

Abstract

In this paper, we propose an Arabic Question-Answering (Q-A) system called QASAL «Question –Answering system for Arabic Language». QASAL accepts as an input a natural language question written in Modern Standard Arabic (MSA) and generates as an output the most efficient and appropriate answer. The proposed system is composed of three modules: A question analysis module and Semantic Search module and answer extraction module.

Introduction

The amount of available information is becoming very huge, especially with the Web proliferation. The problem faced by the user is not the lack of documents or information but is the lack of time to find a short and precise answer among the variety of available documents. So, the information precision became of a great importance. Information Retrieval (IR) systems are designed to retrieve the documents which are estimated to be relevant to the user's query. Search engines, for example, offer a lot of links toward Web pages, but are not able to provide an exact answer. Therefore, these systems are unable to satisfy the users who are interested in obtaining a simple answer to a specific question [4]. Thus, a new need is emerged: the possibility of obtaining a unique, brief and concise answer. It is the main goal of Q-A systems, which aim to retrieve small pieces of texts that contain the answer to the question rather than the list of documents, traditionally returned by search engines. We begin this paper with an overview of the system.

Requirements

- Get question from user.
- Part of speech.
- Question type.
- Stem keywords.
- Semantic search(**Ontology**).
- Answer extraction.

Tasks

- **Analysis**:
    - Part of speech
    - Stemmer
    - Ontology
    - Answer extraction
- **Design**:
    - Interface
- **Implementation**:
    - Part of speech
    - Question type
    - Get keywords
    - Keyword stemming
    - Query retrieval
    - Semantic search
    - Answer extraction
- **Test**:
    - Part of speech
    - Stemmer
    - Semantic search
    - Answer extraction

Tools

- **NetBeans IDE Java**
- **Shereen Khoja Stemmer**
- **Stanford POS-Tagger 2016**
- **Dom XML Parser**
- **Apache Lucene**

Problems

- **Khoja stemmer:**
    - Occasionally a produces a faulty stem when stemming a strange word instead of returning “NOT STEMMED”. The problem could be easily addressed by adding more strange words to the strange words file included in the stemmer corpus.

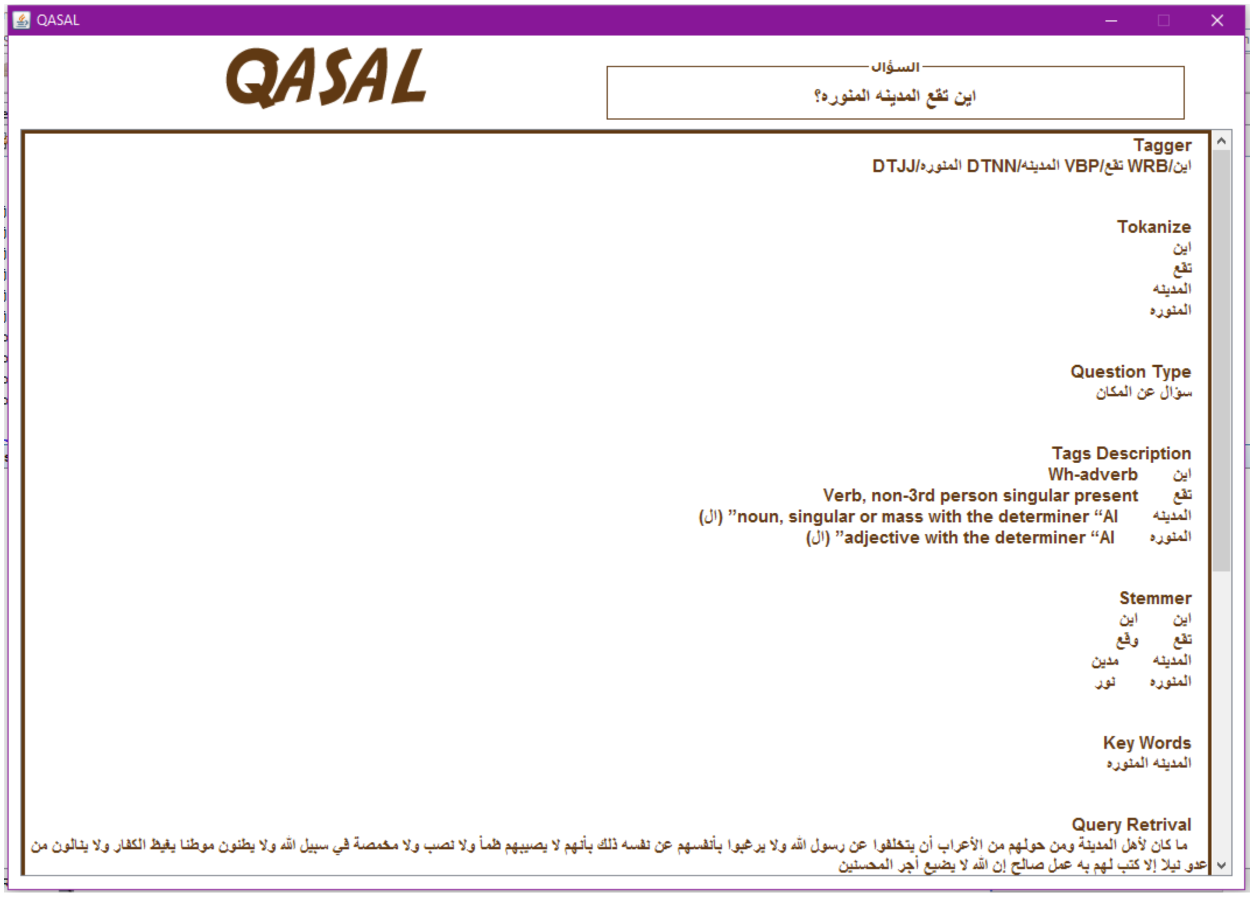
- **Ontology:**
    - Reading Arabic owl/xml files is a bit different and produces slightly difference results than owl/rdf files.
    - For reading owl/xml files, OWL-API & For owl/rdf files, Apache Jena with SPARQL queries for extracting results and because these two API are complex and difficult we use DOM API for reading OWL Files its easy and fast.

- **Apache Lucene:**
    - Well. The java-based search engine works well. Too well actually, often returning search results that are only distantly related to the query. We also found a difficulty in searching using an expanded query using ontology files.

Execution

First user enters his question about places in **Quran**. Then **QASAL** take his question and analyze it, system uses **Stanford POS-Tagger 2016** for tagging the question to simplify gathering the question type then system search in questions tags set to figure out what is the type of this question, then system removes stop words to simplify gathering the keywords that will help finding the optimal answer, and next to removing stop words system return the nouns and adjectives as the keywords.

Now we have two important things the question type and the keywords, system uses **Shereen Khoja Stemmer** to stem keywords to use it in getting the related query from Quran using **Apache Lucene** to search in **Quran** indexed files, and to search for results then search in **OWL** (**Ontology Web Language**) files that extended from **xml/rdf** files that contain instance of real objects and the relations between them and graph these relation as tree of relations that contain parents and children. System uses **DOM Xml Parser** to search in these files to get the answer.



اين تقع المدينه المنوره؟

Tagger
اين/WRB تقع/VBP المدينه/DTNN المنوره/DTJJ

Tokanize
اين
تقع
المدينه
المنوره

Question Type
سؤال عن المكان

Tags Description

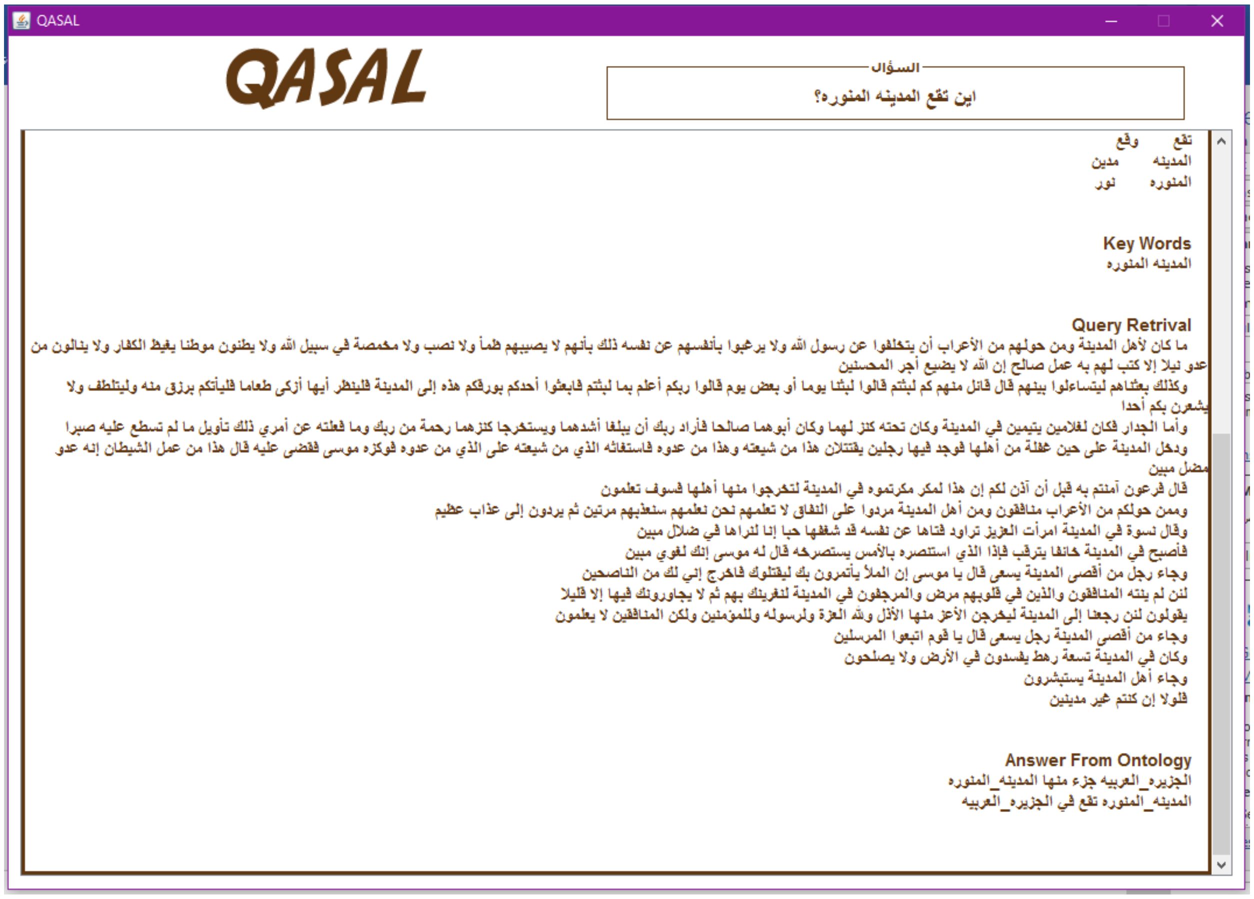
اين	Wh-adverb
تقع	Verb, non-3rd person singular present
المدينه	noun, singular or mass with the determiner “Al” (ال)
المنوره	adjective with the determiner “Al” (ال)

Stemmer

اين	اين
تقع	وقع
المدينه	مدين
المنوره	نور

Key Words
المدينه المنوره

Query Retrival
ما كان لأهل المدينة ومن حولهم من الأعراب أن يتخلفوا عن رسول الله ولا يرغبوا بأنفسهم عن نفسه ذلك بأنهم لا يصيبهم ظمأ ولا نصب ولا مخمصة في سبيل الله ولا يطئون موطئا يغيظ الكفار ولا ينالون من عدو نيلا إلا كتب لهم به عمل صالح إن الله لا يضيع أجر المحسنين



اين تقع المدينه المنوره؟

تقع	وقع
المدينه	مدين
المنوره	نور

Key Words
المدينه المنوره

Query Retrival
ما كان لأهل المدينة ومن حولهم من الأعراب أن يتخلفوا عن رسول الله ولا يرغبوا بأنفسهم عن نفسه ذلك بأنهم لا يصيبهم ظمأ ولا نصب ولا مخمصة في سبيل الله ولا يطئون موطئا يغيظ الكفار ولا ينالون من عدو نيلا إلا كتب لهم به عمل صالح إن الله لا يضيع أجر المحسنين
وكذلك بعثناهم ليتساءلوا بينهم قال قائل منهم كم لبثتم قالوا لبثنا يوما أو بعض يوم قالوا ربكم أعلم بما لبثتم فابعثوا أحدكم بورقكم هذه إلى المدينة فلينظر أيها أزكى طعاما فليأتكم برزق منه وليتلطف ولا يشعرن بكم أحدا
وأما الجدار فكان لغلامين يتيمين في المدينة وكان تحته كنز لهما وكان أبوهما صالحا فأراد ربك أن يبلغا أشدهما ويستخرجا كنزهما رحمة من ربك وما فعلته عن أمري ذلك تأويل ما لم تسطع عليه صبرا
ودخل المدينة على حين غفلة من أهلها فوجد فيها رجلين يقتتلان هذا من شيعته وهذا من عدوه فاستغاثه الذي من شيعته على الذي من عدوه فوكزه موسى فقضى عليه قال هذا من عمل الشيطان إنه عدو مضل مبين
قال فرعون آمنتم به قبل أن آذن لكم إن هذا لمكر مكرتموه في المدينة لتخرجوا منها أهلها فسوف تعلمون
وممن حولكم من الأعراب منافقون ومن أهل المدينة مردوا على النفاق لا تعلمهم نحن نعلمهم سنعذبهم مرتين ثم يردون إلى عذاب عظيم
وقال نسوة في المدينة امرأت العزيز تراود فتاها عن نفسه قد شغفها حبا إنا لنراها في ضلال مبين
فأصبح في المدينة خائفا يترقب فإذا الذي استنصره بالأمس يستصرخه قال له موسى إنك لغوي مبين
وجاء رجل من أقصى المدينة يسعى قال يا موسى إن الملأ يأتمرون بك ليقتلوك فاخرج إني لك من الناصحين
لئن لم ينته المنافقون والذين في قلوبهم مرض والمرجفون في المدينة لنغرينك بهم ثم لا يجاورونك فيها إلا قليلا
يقولون لئن رجعنا إلى المدينة ليخرجن الأعز منها الأذل ولله العزة ولرسوله وللمؤمنين ولكن المنافقين لا يعلمون
وجاء من أقصى المدينة رجل يسعى قال يا قوم اتبعوا المرسلين
وكان في المدينة تسعة رهط يفسدون في الأرض ولا يصلحون
وجاء أهل المدينة يستبشرون
فلولا إن كنتم غير مدينين

Answer From Ontology
الجزيره_العربيه جزء منها المدينه_المنوره
المدينه_المنوره تقع في الجزيره_العربيه

Implementation

Question **POS** is very simple step in this system by using Stanford POS Tagger library that implemented in java and easy to use just call a function **tagString()** that take the a string as a parameter and return it tagged in this form ‘word/tag’.

Question Tokenizing is a simple step too just calling a the function **split()** that take delimiter string as a parameter and return an array of words.

Determine Question Type by search for question tag in the tagged question then check the type of this word if it is ‘اين’ or any other word.

Determine Keywords is the same as the question type but here we search for nouns and adjectives tags and return the words.

The next step is getting the tags description by search in a switch statement to find the tag and return its description, note that tags for Arabic words is less than tags for English words in Stanford POS tagger.

Stem Keywords is a sort of complex step that use a complex algorithm in finding the root if the word depending on some dataset stored in files all that is implemented for us by Shereen Khoja who made a Stemmer that is easy to use but needs to be improved and to increase the dataset to reduce the percent of mistakes.

Query retrieval is the step to return all related query with the highest score from Quran using Apache Lucene that index the Quran in indexed files, that is help it to return the query very fast instead of searching in 6000 file, this step ease the step of extraction answer.

Semantic Search is the step of finding the synonymous of the keywords to increase the border of the answer by searching in ontology files that has an extension ‘OWL’, using DOM xml parser that is easy to implement and use but it is slow little bit.

Extraction answer step in this system is finding the related relation in ontology files like isLocated_in, is_inAreaOf, isMountain_in, is_a, is_partOf and hasPart these are the relation tags in ontology files.

Conclusion

In this paper, we proposed an Arabic Question Answering System called QASAL. QASAL takes advantage of some linguistic techniques from IR and NLP to process a collection of Arabic text documents containing factoid questions as well as their different answers. The overall success of the system is limited to the number of available tools developed for the Arabic language. As perspectives, we intend to extend our system for definition question and to use the Arabic WordNet in order to allow a semantic query expansion.

Reference

- S. Abney, M.Collins and A. Singhal, 2000. “Answer extraction”. In Proceedings of the 6th Applied Natural Language Processing Conference (ANLP-2000), 296–301.
- S. Abuleil, “Extracting Names From Arabic Text For Question-Answering Systems”, Chicago State University.

- https://nlp.stanford.edu/software/tagger.shtml
- http://zeus.cs.pacificu.edu/shereen/research.htm
- https://www.mkyong.com/java/how-to-read-xml-file-in-java-dom-parser/
- https://lucene.apache.org/
- https://lucene.apache.org/core/6_4_2/demo/index.html