

Al-Bayan: An Arabic Question Answering System for the Holy Quran

Heba Abdelnasser

heba.abdelnasser@alex.edu.eg

Maha Ragab

maha.ragab@alex.edu.eg

Bassant Farouk

bassant.farouk@alex.edu.eg

Reham Mohamed

reham.mohmd@alex.edu.eg

Alaa Mohamed

alaa.mohmd@alex.edu.eg

Nagwa El-Makky

nagwa.elmakky@alex.edu.eg

Marwan Torki

marwan.torki@alex.edu.eg

Computer and Systems Engineering Department
Alexandria University, Egypt

Abstract

Recently, Question Answering (QA) has been one of the main focus of natural language processing research. However, Arabic Question Answering is still not in the mainstream. The challenges of the Arabic language and the lack of resources have made it difficult to provide Arabic QA systems with high accuracy. While low accuracies may be accepted for general purpose systems, it is critical in some fields such as religious affairs. Therefore, there is a need for specialized accurate systems that target these critical fields. In this paper, we propose Al-Bayan, a new Arabic QA system specialized for the Holy Quran. The system accepts an Arabic question about the Quran, retrieves the most relevant Quran verses, then extracts the passage that contains the answer from the Quran and its interpretation books (*Tafseer*). Evaluation results on a collected dataset show that the overall system can achieve 85% accuracy using the top-3 results.

1 Introduction

Nowadays, the Web has become the main source of information where lots of terabytes of data are added every day in all fields. With this increase of data on the Web, there is a critical need for advanced search facilities that satisfy users' demands with high accuracy. This leads to several problems: the first problem is that most of the available search engines provide users with documents that are relevant to their demands; however, the users should take the trouble of searching for the answers inside each document. This increased the need for

Question Answering (QA) systems that provide the users with direct answers to their questions. While great efforts have been made to provide reliable QA systems for different languages, very few attempts have been made to investigate QA for the Arabic language.

The second problem is the quality of the data. The development of social networks made the users not only encouraged to search on the Web but also to post their opinions and knowledge. Although this is an advantage for sharing knowledge in different fields and massively increasing the data on the Web, it is critical for religious affairs where users may post untrusted or false information. Observing the Arabic Web, we found that this problem is very common for the Holy Quran, where large amount of incorrect data is published on different sites which may provide a spurious view of the Islamic religion.

The third problem is the challenges of the Arabic language. **Arabic is highly inflectional and derivational**, which makes its morphological analysis a complex task. Derivational: where all the Arabic words have a three or four characters root verbs. Inflectional: where each word consists of a root and zero or more affixes (prefix, infix, suffix). **Arabic is characterized by diacritical marks (short vowels)**, the same word with different diacritics can express different meanings. Diacritics are usually omitted which causes ambiguity. **Absence of capital letters in Arabic** is an obstacle against accurate named entities recognition. Finally, **the lack of Arabic resources**, such as corpora, makes Arabic NLP research more challenging.

In this paper, we propose our solutions to these problems. We introduce Al-Bayan: a new Ara-

bic QA system specialized for the Quran. Al-Bayan aims at understanding the semantics of the Quran and answering users questions using reliable Quranic resources. Mainly, we use the Quran and its interpretation books (*Tafseer*) of trusted Quranic scholars as our sources of information. Our main contribution can be summarized in the following points:

1. Building a Semantic Information Retrieval module that retrieves the semantically related verses to user's questions.
2. Increasing the accuracy of question analysis by applying a highly accurate Arabic tool for morphological analysis and disambiguation and by using a state of the art classifier, i.e. Support Vector Machine (SVM) to classify questions.
3. Extracting the ranked answers to the input questions from the retrieved verses and their interpretation with high accuracy.

The rest of the paper is organized as follows: Section 2 shows some of the work related to our system. Section 3 shows the details of the system model. Section 4 shows the datasets that we used to build the system. In Section 5, we show some of the initial results. Finally, we conclude the paper and give directions to future work in Section 6.

2 Related Work

Our work is related to prior work in both Quranic research and Question Answering systems.

(a) Quranic Research: Several studies have been made to understand the Quranic text and extract knowledge from it using computational linguistics. Saad et al. (2009) proposed a simple methodology for automatic extraction of concepts based on the Quran in order to build an ontology. In (Saad et al., 2010), they developed a framework for automated generation of Islamic knowledge concrete concepts that exist in the holy Quran. Qurany (Abbas, 2009) builds a Quran corpus augmented with a conceptual ontology, taken from a recognized expert source 'Mushaf Al Tajweed'. Quranic Arabic Corpus (Atwell et al., 2011) also builds a Quranic ontology of concepts based on the knowledge contained in traditional sources of Quranic analysis, including the sayings of the prophet Muhammad (PBUH), and the *Tafseer* books. Khan et al. (2013) developed a simple ontology for the

Quran based on living creatures including animals and birds that are mentioned in the Quran in order to provide Quranic semantic search. AlMaayah et al. (2014) proposed to develop a WordNet for the Quran by building semantic connections between words in order to achieve a better understanding of the meanings of the Quranic words using traditional Arabic dictionaries and a Quran ontology.

Other attempts for text-mining the Quran were proposed such as: QurAna (Sharaf and Atwell, 2012) which is a corpus of the Quran annotated with pronominal anaphora and QurSim (Sharaf and Atwell, 2012) which is another corpus for extracting the relations between Quran verses.

b) Question Answering (QA) Systems: Although a large number of QA systems were proposed for the English language such as the work proposed by Fleischman et al. (2003), Ittycheriah and Roukos (2006), Kaisser (2012), the Arabic QA research is still limited in terms of accuracy. Some Arabic systems have been proposed such as: QARAB (Hammo et al., 2002) which is a QA system that takes factoid Arabic questions and attempts to provide short answers. ArabiQA (Benajiba et al., 2007) which is fully oriented to the modern Arabic language. It also answers factoid questions using Named Entity Recognition. However, this system is not completed yet. DefArabicQA (Trigui et al., 2010) presents a definitional QA system for the Arabic language. Arabic QA4MRE (Trigui et al., 2012) introduced the Arabic language for the first time at CLEF. This system proposed a new approach which can answer questions with multiple answer choices from short Arabic texts. However, its overall accuracy is 0.19. Also, all these systems target the modern standard Arabic. To the best of our knowledge, no previous research was proposed for the Quranic classical Arabic question answering.

3 System Model

Al-Bayan system architecture is shown in Figure 1. The input question passes mainly through three stages. The first stage is Question Analysis, where the input question is preprocessed and classified to get the expected answer type. The preprocessed question then enters the second stage, Information Retrieval. In this stage, the semantically relevant verses are retrieved using offline preprocessed Quranic data. Finally, the expected answer type and the retrieved verses are fed to the Answer Ex-

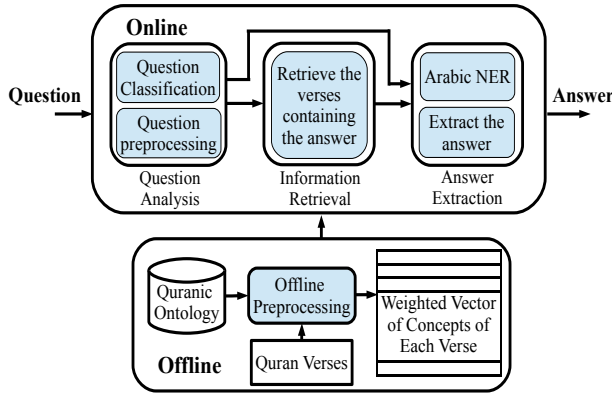


Figure 1: System Architecture

traction module which extracts the answer from the obtained verses and their *Tafseer* using a set of features. We first present the preprocessing operations that are used in online and offline phases. Then, we present the different modules of the system.

3.1 Preprocessing Operations

Text preprocessing is done by applying morphological analysis to identify the structure of text such as: morphemes, roots, affixes, stems, part of speech (POS) tags, etc. The Arabic language is, morphologically, one of the most complex and rich languages. Moreover, the Quranic Arabic is morphologically more complex, since each word may have more than one meaning and a word may have more than one POS tag. Also, the Arabic text of the Quran is fully diacritized, while most of the questions are written without diacritics. For preprocessing, we used MADA (Morphological Analysis and Disambiguation for Arabic) (Habash et al., 2009) which is one of the most accurate Arabic preprocessing toolkits. MADA can derive extensive morphological and contextual information from raw Arabic text, and then use this information for high-accuracy part-of-speech tagging, diacritization, lemmatization, disambiguation, stemming, and glossing in one step. Each term in the input text will be represented by its stem and POS tag, in the following format (stem:POS) using Buckwalter transliteration (Buckwalter, 2002). We remove pronouns, prepositions, conjunctions and other POS types, since these words are stopwords and must not affect the information retrieval indexing. In our system, we apply MADA preprocessing in two different phases: on the Quran and its *Tafseer* in the offline phase, and on the input question in the

online phase.

3.2 Question Analysis

The system first takes the Arabic question which is preprocessed to extract the query that will be used in the Information Retrieval module. The question is also classified to get the type of the question, and consequently the type of its expected answer, which will then be used in the Answer Extraction module.

3.2.1 Question Preprocessing

The preprocessing operations discussed in Section 3.1 are applied to the input question. The preprocessed question is represented by a vector of terms where each term consists of a stem and a POS tag.

3.2.2 Question Classification

We classify the question to the anticipated type of the answer. This information would narrow down the search space to identify the correct answer. The most straight forward question classification is the Rule-based approach; where a set of rules is used to derive the answer type (for example: the answer of Who/Whom is of type person). The derivation of expected answer types is often carried out by means of machine learning approaches, such as the work of Li and Roth (2002). This task relies on three parts: taxonomy of answer types into which questions are to be classified, a corpus of questions prepared with the correct answer type classification, and an algorithm that learns to make the actual predictions given this corpus. We use an SVM classifier for this purpose and construct its training data. We also introduce a new taxonomy built specially for our system. More details about our dataset and taxonomy are mentioned in Section 4.

Unlike Rule-based classifiers, our SVM classifier can classify questions in which the question word is omitted. For example the two questions: (Where did Allah talk to Moses?) and (What is the name of the mountain at which Allah talked to Moses?), both have the same answer type (Location). However, the Rule-based classifier cannot determine the correct answer type of the second question since the question word (Where) is omitted. Our SVM classifier, on the other hand, learns that a mountain name is of type location, therefore it correctly classifies the two questions.

3.3 Information Retrieval (IR)

The preprocessed question is now fed to the Information Retrieval module that retrieves the most semantically related verses from the Quran and its interpretation books (*Tafseer*). Our approach is based on the explicit semantic analysis approach (Gabrilovich and Markovitch, 2007) that augments keyword-based text representation with concept-based features, automatically extracted from massive human knowledge repositories such as Wikipedia. However, instead of using Wikipedia as ontology, we build our Quranic ontology of concepts which classifies the Quran verses according to their topics. Details of building our Quranic ontology are shown in Section 4. We use machine-learning techniques to build a Semantic Interpreter as in (Gabrilovich and Markovitch, 2007) that maps fragments of natural language text into a weighted vector of Quranic concepts. Each leaf concept in the ontology has a list of verses, which are related to this concept. For each leaf concept C_i , a document D_i is constructed, where D_i is a document of verses and their *Tafseer* that belong to C_i . Then preprocessing on D_i is applied and finally an index on D_i is created using Lucene Indexer¹. Each Quranic concept will be represented by a vector of terms that occur in the corresponding document. Entries of this vector are assigned weights using the TFIDF scheme. These weights quantify the strength of association between terms and concepts. To speed semantic interpretation, we build an inverted index which maps each term into a list of concepts in which it appears. Using the Semantic Interpreter in a way similar to that in (Gabrilovich and Markovitch, 2007), a weighted vector of concepts is generated for each verse in the Quran and stored in our database. This is done in the offline phase. Similarly, the vector of the input query is calculated in the online phase. To select the top-scoring verses that are semantically related to the user question we compute the cosine similarity between the concept vector of the input query and the concept vector of each verse in the Quran.

3.4 Answer Extraction

After the relevant verses are retrieved, these verses, their *Tafseer* and the expected answer type are fed into the Answer Extraction stage to extract the final answer to the input question. We define the answer as the phrase which contains the expected answer

type (a named entity or a description of a named entity). The Answer extraction stage consists of the following steps: First, the named entities in the input question are identified. Then, several features are extracted which are used to rank each candidate answer.

3.4.1 Arabic Named Entity Recognition

Named Entity Recognition (NER) is a subtask of information extraction, where each proper name in the input passage - such as persons, locations and numbers - is assigned a named entity tag. We build the training data as shown in Section 4, then use it to feed LingPipe tool² which constructs the NER model. The NER model is then used in the online phase to tag the input text.

3.4.2 Feature Extraction

Once we have the preprocessed question Q tagged with named entities, we divide the relevant verses and their *Tafseer* into passages such that each passage is a candidate answer. For each candidate answer A , we get the probability of correctness C given the question Q and the candidate answer A . Then, the few candidate answers that have the highest probability of correctness are returned. A set of features are used to calculate the probability of correctness as mentioned by (Wang, 2006), such as:

- (a) Maximum number of matched words between the input question and the candidate answer.
- (b) The type of the question's expected answer if it matches with the extracted named entity in the answer passage in case of factoid questions.
- (c) Is-A relationship in case of definitional questions, in the form: 'NE' is a 'description'.
- (d) The maximum count of named entity types that occurred in the question occurring in the candidate answer.
- (e) The minimum distance between matched terms in the passage.

4 Datasets

In this section, we describe the datasets that we used in different modules of the system.

¹<http://lucene.apache.org/>

²<http://alias-i.com/lingpipe/>

Quranic Ontology and Tafseer Books:

We integrated the Quranic Corpus Ontology (Atwell et al., 2011) and the Qurany Ontology (Abbas, 2009), to form the Quranic conceptual ontology that we use in our system. The **Quranic Corpus Ontology** uses knowledge representation to define the key concepts in the Quran, and shows the relationships between these concepts using predicate logic. The **Qurany Ontology** is a tree of concepts that includes all the abstract concepts covered in the Quran. It is imported from 'Mushaf Al Tajweed' list of topics. This integration was difficult since we had to resolve the overlapping between the two ontologies. There were also some mistakes in the Qurany Concept Tree. So, we had to manually revise the 1200 concepts and their verses.

The Holy Quran consists of 6236 verses. In our Quranic ontology, each verse must be classified to one or more concepts depending on the semantics of this verse. After adding Quranic Corpus ontology, there were 621 verses without concepts, so we added them under their most suitable concepts to complete the ontology using a similarity measure module. This module measures the similarity between classified and unclassified verses to determine the concepts of unclassified verses. Now, our final ontology contains 1217 leaf concepts and all verses of the Quran. Under each concept in our ontology, we save the related verses with their *Tafseer*, that is used to build the inverted index. We use two *Tafseer* books: (Ibn-Kathir, 1370) and (Al-Jaza'iri, 1986), which are two of the most traditional books used by Islamic scholars. It is possible to add other books to enrich our corpus data. We also use the *Tafseer* books to extract the candidate answer passages.

NER Data:

To train our NER module, we need a new annotated corpus specialized for the Quran. Fortunately, Quranic Arabic corpus provides NE annotations for the Quran. This corpus is a hierarchical concept tree that has about 14 main classes. We mapped these classes to 5 categories and also manually added a new class for Numbers. We used a book called 'Numbers and Ratios in Quran' (Ali, 2008) to tag the numbers in the Quran. Table 1 shows the final classes and their members.

El O	0	وإذ
Abogado B-PER	0	واعدنا
General I-PER	B-creation	موسى
del I-PER	B-number	أربعين
Estado I-PER	0	ليلة
, O	0	ثم
Daryl B-PER	0	اتخذتم
Williams I-PER	0	العجل
, O	B-entity	

(a) CoNLL 2002

(b) Al-Bayan

Figure 2: Format of the NER training file. Each named entity is tagged with its beginning or continuing token picked out with tags B-class and I-class respectively. If the word is not named entity it is tagged with 0.

Our training data was annotated to have the same format of CoNLL 2002 corpora³ as shown in Figure 2.

Question Classification Data:

We built a new taxonomy for Question Classification based on the NE categories discussed above. We also had to construct the training and test data suitable for this taxonomy. Our data consists of 230 classified questions collected randomly from forums or some common Quranic questions, divided into 180 questions used for training and 50 questions used for testing. The questions are classified according to their answer types into: (Creation, Entity, Physical, Location, Number, Description), where the first 5 classes are the named entities detected by the NER module, and the last class discriminates the definitional questions. The distribution of the questions among these classes is shown in Table 2

5 Evaluation

We evaluated the different modules of our system as well as the overall system accuracy.

5.1 NER Module

We evaluated this module using LingPipe evaluator. The training data is divided into 3 folds and the overall Precision, Recall and F-measure are calculated. Results are shown in Figure 3.

³<http://www.cnts.ua.ac.be/conll2002/ner/>

Al-Bayan classes	NER	Members
Creation		Human - Angels - jinn. E.g. Muhammed, Jibreel and Satan
Location		After life locations - Geographical locations - Worship locations. For example, the heaven, Mosque, and Church
Entity		Events - Holy books - Languages - Religions - False deity - Organics. For example, Day of Resurrection, Quran, Injeel, Arabic, Islam, Christianity and Idol and (Bone)
Physical Entity		Astronomical Body - Artifact - Weather Phenomena - Physical Substance. For example, the Sun, Earth, (Boat), Rain, and Dust
Numbers		One, Two,...

Table 1: NER classes

Class	Creation	Entity	Physical	Location	Number	Description
Questions	90	40	17	22	14	45

Table 2: Distribution of the question classification data.

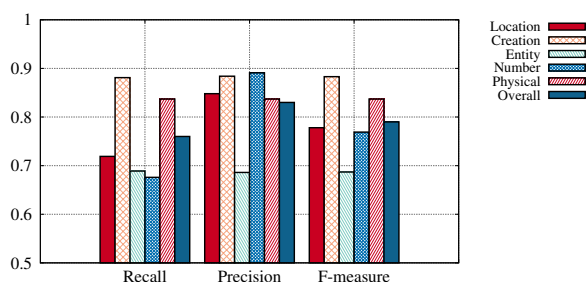


Figure 3: Quranic Arabic NER results

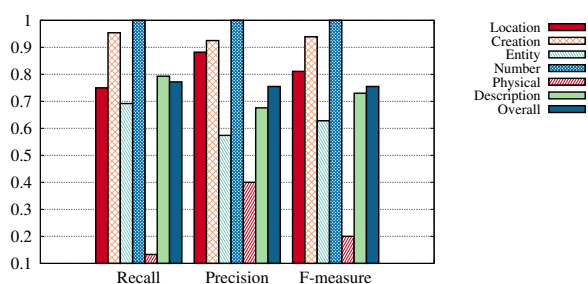


Figure 4: Question classifier results.

5.2 Question Classification Module

We evaluated the classifier based on our proposed taxonomy using 230 Arabic questions. We used 180 questions for training. The overall accuracy of the classifier using 3-folds cross-validation is 77.2%. The precision, recall and F-measure of the 6 classes is shown in Figure 4. We also evaluated the classifier using an independent set of 50 questions. The accuracy of the classifier on this set is 86%.

5.3 Overall System Evaluation

Evaluating our overall system is not an easy task, since we do not have a gold-standard for the Quran questions to compare with our results. Humans have the ability to judge the semantic relatedness of texts. Human judgments can be considered a gold standard against which computer algorithms are evaluated. Therefore, we asked some experts in Quran to judge our system accuracy. The system was evaluated by 5 Quran experts, using 59 questions. The output of our system for each question was the top-3 answers and the top-5 related verses. Each expert marked each verse or answer as right or wrong.

Figure 5 shows some examples of the evaluation questions with the answers retrieved by Al-Bayan system. For the first question (Who is the Queen of Sheba?), although the answer (Bilkis) is not explicitly mentioned in the Quran, the system was able to extract the correct answer from the Tafseer of the related verses. For the second question (How many months is the period of waiting of widows?), the system elegantly extracts the complete answer which includes different conditions of the pregnant and non-pregnant widows. The third and fourth questions are examples of definitional questions.

We used the TopN accuracy (Manning et al., 2008) to evaluate the overall system. TopN accuracy of correct answers is calculated as the number of questions in which at least one of the top N answer candidates is correct, divided by the total number of questions. We also calculate the preci-

	IR Module		Overall System
Top-1	0.692	Top-1	0.650
Top-5	0.847	Top-3	0.854
Precision	0.57	Precision	0.73

(a)

(b)

Table 3: Experts Evaluation Results

sion when the system outputs 5 related verses and 3 answer passages. Table 3a shows the results of the verses retrieved from the IR module and Table 3b shows the results of the overall system. We notice that the Top-3 results of the overall system is better than Top-1 results, that is why we return Top-3 answers to the user to increase the probability of correct answers. We also noticed that the results of the overall system is better than information retrieval results, which shows that answer extraction module improves the accuracy of the overall system.

6 Conclusion and Future Work

In this paper, we proposed a novel Question Answering system for the Quran, that takes an Arabic question as an input and retrieves semantically relevant verses as candidate passages. Then an answer extraction module extracts the answer from the retrieved verses accompanied by their Tafseer. We also proposed a new taxonomy for Quranic Named Entities and constructed an Arabic Question Classifier based on state-of-the-art techniques. Our initial results evaluated by Quranic experts show the feasibility of constructing an accurate QA system specialized for the Quran.

In the future, we plan to explore more complex questions such as: list-type questions. In order to improve the accuracy of the system, we plan to use active learning techniques which are appropriate when the gold-standard is scarce or expensive to obtain. Thus, Quran experts can give their feedback about the answers and the system would learn from this feedback and improve its results. Finally, we plan to make the proposed system publicly available to the research community.

Acknowledgments

We would like to thank Dr. Noorhan Hassan Abbas and Prof. Dr. Eric Atwell from University of Leeds for providing us with the Quranic Ontology files.

References

- Abdul-Baqee M. Sharaf and Eric Atwell. 2012. *QurAna: Corpus of the Quran annotated with Pronominal Anaphora*. LREC.
- Abdul-Baqee M. Sharaf and Eric Atwell. 2012. *Qur-Sim: A corpus for evaluation of relatedness in short texts*. LREC.
- Abraham Ittycheriah and Salim Roukos. 2006. *IBM's statistical question answering system-TREC-11*. Technical report, DTIC Document.
- Abu Bakr Al-Jaza'iri. 1986. *Aysar al-Tafasir li Kalaam il 'Aliyy il Kabir*.
- Abu Islam Ahmed bin Ali. 2008. *Numbers and Ratios in the Quran*.
- Bassam Hammo, Hani Abu-Salem and Steven Lytinen. 2002. *QARAB: a question answering system to support the Arabic language*. Proceedings of the ACL-02 workshop on Computational approaches to semitic languages.
- Christopher D Manning, Prabhakar Raghavan and Hinrich Schütze. 2008. *Introduction to information retrieval*, volume 1. Cambridge university press Cambridge.
- Eric Atwell, Claire Brierley, Kais Dukes, Majdi Sawalha and Abdul-Baqee Sharaf. 2011. *A An Artificial Intelligence Approach to Arabic and Islamic Content on the Internet*. Proceedings of NITS 3rd National Information Technology Symposium.
- Evgeniy Gabrilovich and Shaul Markovitch. 2007. *Computing Semantic Relatedness Using Wikipedia-based Explicit Semantic Analysis*, volume 7. Proceedings of the 20th international joint conference on artificial intelligence.
- Hikmat Ullah Khan and Syed Muhammad Saqlain and Muhammad Shoaib and Muhammad Sher. 2013. *Ontology Based Semantic Search in Holy Quran.*, volume 2. International Journal of Future Computer and Communication, 570-575.
- Ismail Ibn-Kathir. 1370. *Tafsir al-Qur'an al-Azim*.
- Manal AlMaayah, Majdi Sawalha, and Mohammad AM Abushariah. 2014. *A Proposed Model for Quranic Arabic WordNet*. LRE-REL2, 9.
- Mengqiu Wang. 2006. *A Survey of Answer Extraction Techniques in Factoid Question Answering*, volume 1. Association for Computational Linguistics.
- Michael Fleischman, Eduard Hovy and Abdessamad Echihabi. 2003. *Offline strategies for online question answering: answering questions before they are asked*, volume 1. Proceedings of the 41st Annual Meeting on Association for Computational Linguistics.

<p>من هي ملكة سبأ ؟</p> <p>فَكَتَبَ سُلَيْمَانُ كِتَابًا إِلَى بَلْقَيْسَ مَلِكَةَ سَبَأَ، وَأَمَرَ الْهُدُودَ بِحَمْلِهِ إِلَيْهَا، وَبِالْقَائِهِ بَيْنَ يَدَيْهَا، ثُمَّ أَمَرَهُ بِالتَّنْحِي عَنْهُمْ جَانِبًا لِيُلاحِظَ مَا سَتَفَعَلُهُ بِالكِتَابِ، وَمَاذَا يَكُونُ رَدُّهَا عَلَيْهِ، فَحَمَلَ الْهُدُودُ الْكِتَابَ إِلَيْهَا، وَأَلْقَاهُ بَيْنَ يَدَيْهَا</p>
<p>كم شهر عدة الأرامل ؟</p> <p>يَأْمُرُ اللهُ تَعَالَى النِّسَاءَ اللَّوَاتِي يُتَوَقَّى عَنْهُنَّ أَرْوَاجُهُنَّ بِأَنْ يَعْتَدِينَ أَرْبَعَةَ أَشْهُرٍ وَعَشْرَ لَيَالٍ (وَالْحُكْمُ يَشْمَلُ الرِّجَالَ الْمَدْخُولِ بِهِنَّ وَغَيْرَ الْمَدْخُولِ بِهِنَّ)، وَلَا يَسُدُّ عَنْ هَذِهِ الْحَالَةِ إِلَّا الْمُتَوَقَّى عَنْهَا رَوْجُهَا وَهِيَ حَامِلٌ، فَإِنَّ عِدَّتَهَا تَكُونُ بِوَضْعِ حَمْلِهَا</p>
<p>ما هي عقوبة السارق؟</p> <p>يَأْمُرُ اللهُ تَعَالَى بِقَطْعِ يَدِ السَّارِقِ وَالسَّارِقَةِ، وَكَانَ الْقَطْعُ مَعْمُولًا بِهِ فِي الْجَاهِلِيَّةِ، فَقَرَّرَهُ الْإِسْلَامُ، وَجَعَلَتْ لَهُ شُرُوطًا</p>
<p>ما معنى الجاثية ؟</p> <p>جَاثِيَةٌ بَارَكَةٌ عَلَى الرُّكْبِ، لِشِدَّةِ الْهَوْلِ</p>

Figure 5: Examples of the evaluation questions with the answers retrieved by Al-Bayan system.

- Michael Kaisser. 2012. *Answer sentence retrieval by matching dependency paths acquired from question/answer sentence pairs*. Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics.
- Nizar Habash, Owen Rambow and Ryan Roth. 2009. *MADA+ TOKAN: A toolkit for Arabic tokenization, diacritization, morphological disambiguation, POS tagging, stemming and lemmatization*. Proceedings of the 2nd International Conference on Arabic Language Resources and Tools (MEDAR), Cairo, Egypt.
- Noorhan Hassan Abbas. 2009. *Quran's search for a Concept' Tool and Website*. M. Sc. thesis, University of Leeds (School of Computing).
- Omar Trigui, Lamia Hadrach Belguith and Paolo Rosso. 2010. *DefArabicQA: Arabic Definition Question Answering System*. Workshop on Language Resources and Human Language Technologies for Semitic Languages, 7th LREC, Valletta, Malta.
- Omar Trigui, Lamia Hadrach Belguith, Paolo Rosso, Hichem Ben Amor and Bilel Gafsaoui. 2012. *Arabic QA4MRE at CLEF 2012: Arabic Question Answering for Machine Reading Evaluation*. CLEF (Online Working Notes/Labs/Workshop).
- Saidah Saad, Naomie Salim, and Hakim Zainal. 2009. *Pattern extraction for Islamic concept.*, volume 2. Electrical Engineering and Informatics, ICEEI.
- Saidah Saad, Naomie Salim, Hakim Zainal and S. Azman M. Noah. 2010. *A framework for Islamic knowledge via ontology representation.*. Information Retrieval & Knowledge Management, (CAMP).
- Tim Buckwalter. 2002. *Arabic transliteration*. URL <http://www.qamus.org/transliteration.htm>.
- Xin Li and Dan Roth. 2002. *Learning question classifiers*, volume 1. Proceedings of the 19th international conference on Computational linguistics.
- Yassine Benajiba, Paolo Rosso and Abdelouahid Lyhyaoui. 2007. *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, Morocco.