

Available online at <http://www.mecs-press.net/ijeme>

Developing an Intelligent Question Answering System

Waheeb Ahmed^{a1}, Ajusha Dasan^a, Babu Anto P^a

^a*Kannur University, Kannur, Kerala 670567, India*

Received: 03 January 2017; Accepted: 28 March 2017; Published: 08 November 2017

Abstract

The goal of an intelligent answering system is that the system can respond to questions automatically. For developing such kind of system, it should be able to answer, and store these questions along with their answers. Our intelligent QA (iQA) system for Arabic language will be growing automatically when users ask new questions and the system will be accumulating these new question-answer pairs in its database. This will speed up the processing when the same question (even if it is in different syntactical structure but semantically same) is being asked again in the future. The source of knowledge of our system is the World Wide Web (WWW). The system can also understand and respond to more sophisticated questions that need a kind of temporal inference.

Index Terms: Question Answering System, Artificial intelligence, Information Extraction.

© 2017 Published by MECS Publisher. Selection and/or peer review under responsibility of the Research Association of Modern Education and Computer Science.

1. Introduction

Question Answering (QA) is a growing research field that employs techniques from Information Retrieval (IR), Information Extraction (IE) and Natural Language Processing (NLP) [1]. This emerging technology approximates an IE application in which a user submits a question in natural language and receives the answer as a text fragment and as short as a word or as long as a sentence [2]. Search Engines depend on probability theory and bivalent logic with lack of real world conceptual knowledge, better precision and inference ability. Retrieved, ranked links or snippets are not much effective with big data analysis and web of information [3]. The aim of question answering (QA) system is to automatically generate concise answers to questions phrased in natural language. It produces only the requested information, unlike traditional search engines which retrieve full documents [4].

Since people ask the same questions over and over again, most of the asked questions proved to be frequent.

* Corresponding author. Tel.: 0091-7510134735
E-mail address: waheeb2003aden@yahoo.com

The QA system will demonstrate better performance when these questions along with their answers are stored in Database(DB). The intelligent QA system should enable users to ask questions in a variety of ways. When the user submit a question to the system, the system will look up for the answers in its DB (if the question is repeated), or on WWW if not available in the database. Consequently, the results will be entered into its DB. For example, the question “who invented mobile phone?”¹, ideally an iQA system would respond with “Martin Cooper”. When analyzing the question the system will first look in the DB if the same question has already been asked before(repeated question), it will retrieve the answer from the DB. This usage saves time for the answer extraction module. If the question is not available in the DB, then it will be analyzed by the question processing module and answer will be retrieved from the web by the answer extraction module. Moreover, the question-answer pair will be stored in the system DB.

Majority of QA systems are mainly depending on keywords query. Where users submit the keywords at first, then matching is performed with the QA pairs in background database by the querying function of the system. Since this method is based on keywords matching, it doesn't address the problem that how to make extensive question analysis and natural language comprehension. A QA system is considered intelligent when it has a high degree of understanding of natural language and it can generate the answers autonomously. Despite that, understanding of natural language is still under investigation by a leading-edge technology, but to some extent it can satisfy some degree of semantic understanding. Our iQA system address these issues, firstly, by maintaining a database for repeated questions and becoming computationally inexpensive. Second, it handles questions with temporal expressions that need special analysis. The system will be growing continuously with every new question supplied by users.

In the following sections, we will firstly give a brief overview of the related work on QA systems and the techniques used to implement them. Then, we will show the architecture of our QA system and a description of the function of its modules. Finally, experimental results of the evaluation of our system using questions' set will be presented.

2. Related Work

In [5] an approach to the construction of a QA system that generates short answers to questions provided in the Arabic language has been proposed. The system uses techniques from Information Retrieval and Natural Language Processing to process the user's question and a collection of Arabic text documents as its main source of knowledge (similar systems in [6][7][8]).

Kurdi developed the first web-based Arabic QA system called JAWEB. It receives native Arabic questions from user and analyzes them in the server. The analysis consists of three modules namely: question analysis, passage retrieval, and answer extraction. It looks for and extracts answers from the web, no knowledge base or DB is used in this system[9].

Shouning Qu et. al [10] developed an intelligent QA system for Chinese language to support interaction between the teachers and the students. The questions asked by students are stored in a knowledge base for future use and it provides a way to accumulate knowledge for students. In addition, the system can show the weak points of student's knowledge by analyzing the track of the questions posed by the students, and this helps teacher to change the way of teaching.

Peter Clark et. al [11] developed an approach that is based on the construction of a catalog of domain-independent question types, where each type is coupled with a question template and question-answering method. They also focused on mechanisms by which a domain expert can transfer his/her questions to the system, rather than the reasoning details used within the Knowledge base itself to get the answers.

A question answering (QA) system that learns how to identify and rank answer passages by analyzing both

¹All questions are translated into English for this paper.

questions and their answers (QA pairs) supplied as training data was proposed in [12]. The QA system decomposes the QA task into features of discovering and learning to score potential answer fragments from a corpus and past question-and-answer examples.

A QA system, Text2KB, that uses question answering based on a knowledge base and external text data was proposed by Denis Savenkov and Eugene Agichtein[13]. The proposed system demonstrates that text resources improve question understanding, candidate generation and ranking. It utilizes web search engines, community question answering and a collection of general text document data, to detect question topic entities, match question phrases to KB predicates, and to enrich the features of the candidates derived from the knowledge base.

In [14] a template-based FAQ answering system is developed for restricted knowledge domains with up to few hundred FAQs. It is argued that when the system is optimized for expected queries, the system shows best performance and requires a limited knowledge base.

In [15] an approach for improving the capabilities of QA systems using semantic techniques was proposed. An approach was implemented that takes natural language queries and converts them to Resource Description Framework (RDF) triples and returns relevant answers. They demonstrated that experiment results showed that the proposed technique performs well for single word answers.

Some systems are based on Freebase or DBpedia[16][17]. Some techniques have been developed for classifying questions given to the system into a pre-defined labels and this will help the answer extraction module use the appropriate technique to extract the answer[18][19][20][21].

3. IQA System Architecture

The main challenge is to effectively find relevant information that answer the user's question from the huge amount of information on the web. Our approach is to answer each question with a database containing question-answer pairs of previously asked questions. If the question is new, the answer will be extracted from the web. The system incrementally extends the knowledge base if it receives new questions not asked before. The Question Answering is also based on temporal inference for questions that contains temporal expressions. Our intelligent QA system include the following modules:

- Question processing module
- Temporal Inference module
- Repeated question module
- knowledge search module
- Answer extraction module
- System Management Module

The architecture of the system is illustrated in Fig.1.

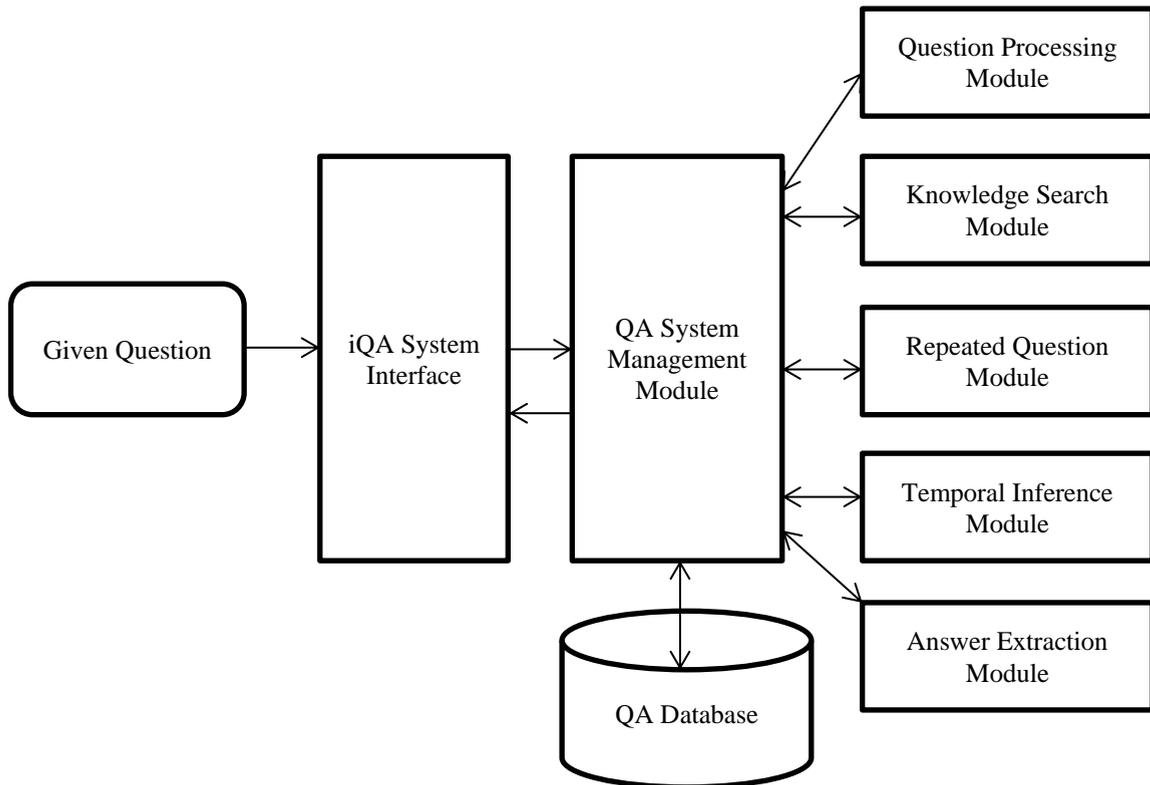


Fig.1. IQA Architecture

3.1. The Question processing module

Question processing module is mainly consisted of three operations: question focus detection, temporal inference, and question classification. As for the task of question classification, the purpose is to analyze the questions to detect the detailed question classes and type of the expected answer in terms of named-entity. Question classification, i.e., classifying the questions into several semantic types, can highly reduce the search space of answers. Question classification accuracy is crucial to the total performance of a question answering system[22]. This information enables the subsequent modules to process documents for accurate extraction of the exact answer from the candidate sentences. Therefore, whenever the system is asked a when-question, it can focus on time or date as candidate answers, a where-question is seeking for location, and a who-question is looking for person, etc.

For question classification subtask, we used a trained classifier from our previous work[23]. However, some questions do not allow the derivation of an expected answer type in the form of named-entity, these questions include which, what why and how questions. In such cases, the question focus can be employed in guiding the next process of answer extraction.

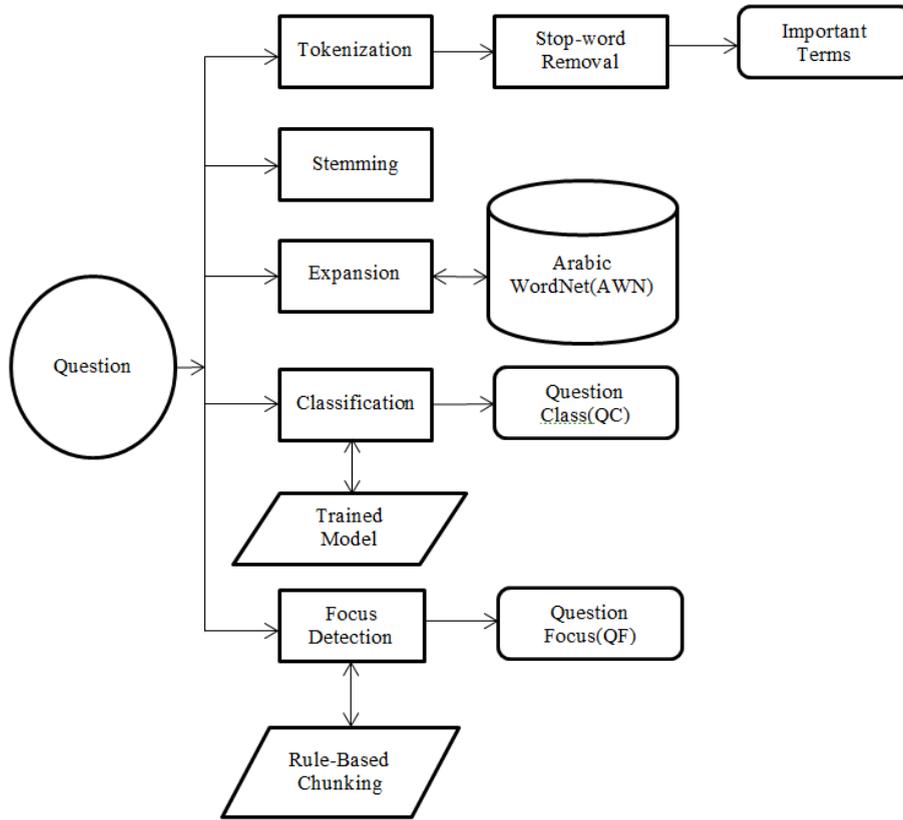


Fig.2. Question Analysis Module

A question focus is defined as a noun phrase in the question that clarifies or disambiguates it and asserts the type of answer being expected. Mostly, the head main noun(head) of the first noun phrase of the question is the question focus. For instance, in question “What movie did Rachel Carson starred in 1962?”, the question focus “movie” can be used to provide supporting indicators to locate the answer in the subsequent process, by seeking for phrases containing the question focus[24]. The different additional NLP pre-processing steps that a question goes through includes stop word removal[25], tokenization [26], stemming [27] where words with less importance are removed from the question, and question expansion[28] where synonyms for some terms of the question are added to improve the information retrieval process. If the question contains any temporal signal, the question will be forwarded to the temporal inference module for further processing. Questions with temporal signals are characterized by (i) the occurrence of a time range expression or a date in the question, (ii) the occurrence of temporal signals such as “after”, “since”, (iii) a question decomposition is needed because of a temporal relation between its events, referred by a temporal signal.

3.2. Temporal inference module

Question answering systems often face temporal information. Identifying relationships between events mentioned in questions is important since many subsequent NLP tasks need these information. By knowing the temporal relationship between two events will help answering a temporal query. To process questions that have temporal inference will depend on (1) the identification of both the events and entities that participate in them, (2) the ordering of these events in the question and also in the document texts, (3) the temporal characteristics

of the entities under question and (4) recognition of the expected answer type and its relations to temporal expressions specified in the question or candidate answers.

When the user asks a question that includes a temporal expression, the answer is supposed to satisfy the temporal constraints. For adding such functionality, QA systems will have to identify relations between temporal expressions and events mentioned in the question and, next, time to depend on temporal inference to justify the answer. To justify answers containing temporal expressions to a question, a way of temporal inference is needed[29].

Table 1. Types of Temporal Questions

Question class	Example
Related-to-Time	“Where can we get information about Space Shuttle since 2000?”
Related-to-Event	“what did president Obama accomplish after he won elections in 2008?”
Related-to-Temporal Order	“Did ancient Greece come before or after Christ?” “what projects did Leonardo da Vinci work on while staying in Milan?”

The answer type of the question “what projects did Leonardo da Vinci work on while staying in Milan?” is given by ANSWER-TYPE-1, and it is an ENTITY. The two predicates “work” and “stay”, which are detected in the given question are represented below along with their arguments:

Temporal signal: While (Arg-Temp)
 Predicate 1: work
 ANSWER-TYPE-1: ENTITY
 Temporal relation: Simultaneous
 Predicate 2: staying
 ANSWER-TYPE-2: TIME RANGE (Arg-Temp)
 Arg0: Leonardo da Vinci
 Arg1: Milan

We used the same argument list in [30]. The arguments of each verbal predicate are given numbers from Arg0 to Arg5. In general, Arg0 refers to agent, Arg1 refers to direct object or theme and Arg2 represents indirect object. For example, in “what projects did Leonardo da Vinci work on while staying in Milan?” when returning the argument structure for the verb-predicate “stay” we find “Arg0:thing” staying and “Arg1:thing” stayed. In addition, some argument may contain functional tags brought from Treebank, e.g. Arg-Dir refers to a directional, Arg-Loc refers to a locative, and Arg-Temp indicates a temporal.

The question contains two Arg-Temp relations. One of them is made between the predicate “work on” and its argument “while staying in Milan”. This argument is starting by the temporal signal “while”, which refers to a temporal relation of equivalency with the implicit TIME RANGE which contains the predicate structure that starts by “stay” occurred. The second Arg-Temp relation is made between the predicate “stay” and the TIME RANGE, which is considered an indirect/implicit answer type for the question. To extract the correct entity as an answer to the question, work relations to “Leonardo da Vinci” must be identified within the TIME RANGE, related to the ANSWER-TYPE-2. Hence, two simpler questions, the first one for ANSWER-TYPE-1(ENTITY) and the second one for ANSWER-TYPE-2(TIME-RANGE), are generated by decomposing the question into two parts. The module that analyze the dependency between the questions(obtained from the decomposition of the original question) specifies that the question related to ANSWER-TYPE-2 requires to be analyzed before analyzing the question related to ANSWER-TYPE-1. Dependencies between the questions also specify the term extractions. Particularly, the terms that are used to return candidate answers for a question are also employed when analyzing any other question it has common dependencies with. The information extracted by this module are sent to the answer extraction module to apply temporal signature inference for each candidate paragraph.

3.3. Repeated question module

This module implements an automatic method for answering repeated questions based on semantic similarity of the question. The method aims at solving one of the significant issues in the information age; i.e. generating answers for questions which are repeatedly asked in a variety of forms. Returned answer along with the question is stored in a database. Repeated question is returned to verify the current question, its semantic features are used to match already answered questions from the database. The repeated question answering module will save resources (for processing the question and finding the answer). For example, “who got Noble Peace Prize in the year 2016” and “name the winner of Noble Peace Prize in 2016”, the answer is same for both questions “Juan Manuel Santos“. The advantage here is that the system will save time from answering questions (which have the same semantics) again and again. When user supplies a question to the system, the system identifies whether the question is being repeated. Our module performs structured and semantic matching. The system has access to the resources available in World Wide Web where it sends the query reformulated from the user question to the search engine like Google and retrieves relevant documents. Questions provided by users will be saved in database. When the question is saved, it will be analyzed and the terms and Named Entities (NEs) will be saved separately. The semantic similarity of questions is done using Cosine similarity [31][32]. The given question undergoes NLP pre-processing (performed by the question processing module).

This module checks whether the question submitted by the user is repeated or not by calculating the similarity between the user question and the questions stored in the DB using Cosine similarity measure.

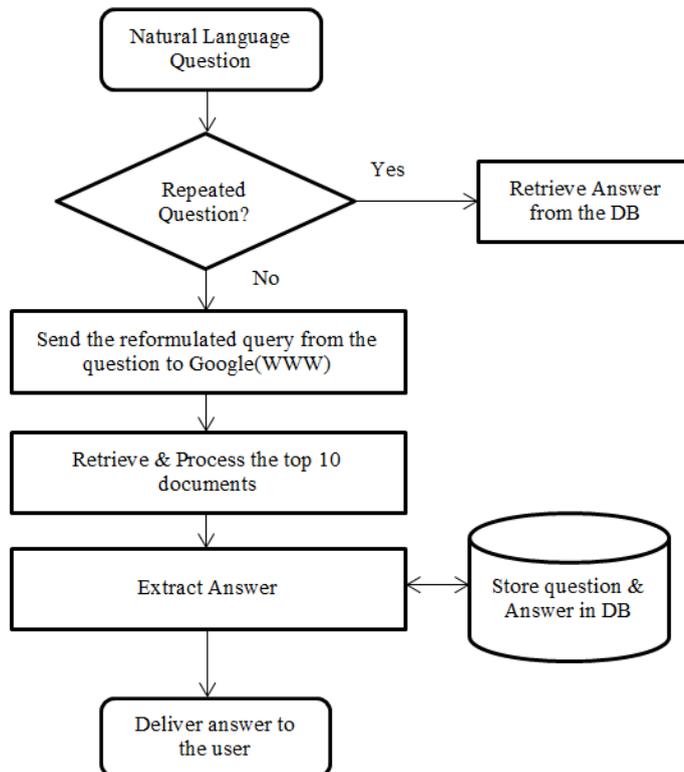


Fig.3. IQA Work Flow

This similarity is calculated using the following formula, given two strings(question & sentence) t_a and t_b , their cosine similarity is calculated as,

$$SIM_C(t_a, t_b) = \frac{t_a \cdot t_b}{|t_a| |t_b|} \tag{1}$$

t_a and t_b are m-dimensional vectors over the term set $T = \{t_1, \dots, t_m\}$. Each dimension represents a term with its weight in the string, which is non-negative value. As a result, the cosine similarity is positive value and bounded between [0,1].

3.4. The answer extraction module

The main role of this module is to retrieve data from the Web based on the search terms extracted from the question (as a part of the output of the question analysis module), and then to select sentences that possibly contain an answer from the returned data for further processing. Our iQA uses Google search engine, as information retrieval back-end, and answer extraction process in which post-processing procedures to extract the potential answers is carried out.

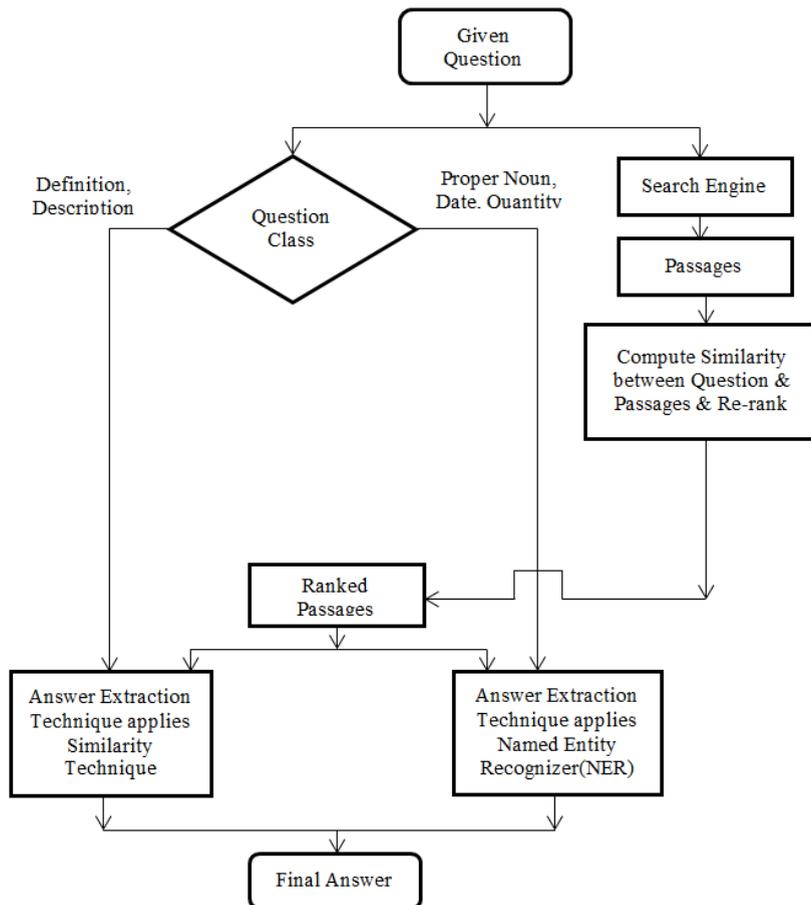


Fig.4. Answer Extraction Module

The returned passages are split into individual sentences. In iQA, sentences that contain the question terms are selected to be passed to the answer extraction module. Using the clues extracted from question classification and focus identification of the question processing module, and the candidate sentences returned by the document processing module, the answer extraction module is ready to proceed. The main goal of this module is the extraction of concise potential answers.

If an expected answer type, derived during the question analysis module, is in the form of named-entity, the candidate sentences retrieved from the document analysis module are first analyzed by a named-entity recognition tool in order to extract phrases of expected answer type, which as result become the candidate answers.

For answers of class “Definition/Description” Cosine similarity(semantic similarity) is calculated between the question terms and the re-ranked passages , the answer with the highest similarity score is retrieved. The answer extraction module depends on the identification of question focus or any other clues obtained during the question analysis module. These candidate answers are then scored based on the similarity with the question focus. Candidate answers with highest score are retrieved. Using data redundancy to validate the accuracy of answer, i.e., the multiple occurrences of the same answer in different documents lend credibility to the proposed answer.

3.5. Knowledge search module

Our QA system is a Web-based Intelligent QA system, i.e., not restricted to any specific domain. The user can supply questions from any knowledge domain. The question will be processed by the question processing module and a query will be reformulated from the extracted terms of the question, this query will be sent to Google search engine to fetch the documents that are likely to contain the answer from the WWW. The top 10 documents are selected for further processing by the AE module. Once the candidate answer is selected by the AE module, the system will automatically add the question and its answer to the database of QA system.

3.6. System management module

The function of this module is to administer the whole system. It will co-ordinate the work of the other modules in the system by managing the pipeline procedure through which the question passes from the moment the user submit it until the final answer is generated.

4. Evaluation and Conclusion

We developed our methodologies in implementing iQA (an intelligent question answering system). Our system mainly consists of 6 modules, namely, question processing module, temporal inference module , repeated question module, knowledge search module , answer extraction module and system management module. In addition, a DB that stores questions and their answers in case the question is being asked in the future. Question analysis module takes users’ natural language questions as input, and classifies the original questions into several types with the help of a training model. The questions supplied by users are transformed into queries to be sent to general-purpose search engines. We used Google as our only backend search engine in our implementation. To deal with questions containing temporal expressions, we develop temporal inference module for this purpose.

At last, the answer extraction module will attempt to look for potential answers for the user’s question. For the purpose of evaluating the general performance of iQA, we used translated version of 200 questions from TREC-10[33] to the iQA. We used the same evaluation technique of TREC, the accuracy of the system to generate correct answers is calculated by the Mean Reciprocal Rank (MRR) which is calculated using the following formula,

$$\text{MRR} = \frac{1}{n} \sum_{i=1}^n \frac{1}{r_i} \quad (2)$$

Where, n refers to the number of the questions to be tested and R is referring to the rank of the correct answer. In case none of the answers retrieved are correct, or if the system does not retrieve any answer, the precision score for that question will be equal to zero. The overall score of the system is measured as the MRR for all the 200 questions. The evaluation result using the MRR is 0.53. Since iQA has a particular function to handle questions with temporal expressions, in addition to using the translated TREC questions as a general evaluation basis, questions with temporal expressions are also used to test the performance of our iQA.

References

- [1] Deepa Yogish, Manjunath T. N., and Ravindra S. Hegadi, "A Survey of Intelligent Question Answering System Using NLP and Information Retrieval Techniques", *International Journal of Advanced Research in Computer and Communication Engineering*, Vol. 5, pp. 536-540, 2016.
- [2] Sanada M. Harabagiu, Steven J. Maiorano, and Marius A. Pasca, "Open-Domain Textual Question Answering Techniques", *Natural Language Engineering Journal*, Vol. 9, pp. 231 - 267, 2003.
- [3] Aniket D Kadam , Shashank D. Joshi , Sachin V. Shinde , and Sampat P. Medhane, "Question Answering Search engine short review and road-map to future QA Search Engine", In *Proceedings of the International Conference on Electrical, Electronics, Signals, Communication and Optimization (EESCO)*, pp. 1-8, 2015.
- [4] Tiansi Dong, Ulrich Furbach, Ingo Glockner and Bjorn Pelzer, "A Natural Language Question Answering System as a Participant in Human Q&A Portals", In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence IJCAI'11*, Vol. 3, pp. 2430-2435, 2011.
- [5] Ghassan Kanaan, Awni Hammouri, Riyad Al-Shalabi and Majdi Swalha, "A New Question Answering System for the Arabic Language", *American Journal of Applied Sciences*, Vol. 6, pp. 797-805, 2009.
- [6] Wissal Brini, Mariem Ellouze , Slim Mesfar and Lamia Hadrich Belguith, "An Arabic question-answering system for factoid questions", In *Proceedings of International Conference on Natural Language Processing and Knowledge Engineering(NLP-KE 2009)*, 2009.
- [7] Bassam Hammo, Hani Abu-Salem, Steven Lytinen, "QARAB: a question answering system to support the Arabic language", In *Proceedings of the ACL-02 workshop on Computational approaches to semitic languages SEMITIC '02*, pp. 1-11, 2002.
- [8] Sman Bekhti, and Maryam Al-Harbi, "AQuASys: A Question-Answering System For Arabic", *International Journal of Academic Research*, Vol. 3, pp.45-54, 2011.
- [9] Heba Kurdi, Sara Alkhaider, and Nada Alfaihi, "Development and Evaluation of a Web Based Question Answering System for Arabic Language", *Fourth International conference on Computer Science & Information Technology*, pp. 187–202, 2014.
- [10] Shouning Qu, Bing Zhang, Xinsheng Yu and Qin Wang, "The Development and Application of Chinese Intelligent Question Answering System Based on J2EE Technology", In *Proceedings of the 1st international conference on Forensic applications and techniques in telecommunications, information, and multimedia and workshop e-Forensics '08*, No. 41, pp. 254-259, 2008.
- [11] Peter Clark, Vinay Chaudhri, Sunil Mishra, Jerome Thomere, Ken Barker and Bruce Porter, "Enabling Domain Experts to Convey Questions to a Machine: A Modified, Template-Based Approach", In *Proceedings of Second International Conference on Knowledge Capture k-Cap'03*, pp. 13-19, 2003.
- [12] Ganesh Ramakrishnan, Soumen Chakrabarti, Deepa Paranjpe and Pushpak Bhattacharyya, "Is Question Answering an Acquired Skill?", In *Proceedings of the 13th international conference on World Wide Web WWW '04*, pp. 111-120, 2004.

- [13] Denis Savenkov, and Eugene Agichtein, "When a Knowledge Base Is Not Enough: Question Answering over Knowledge Bases with External Text Data", In Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval SIGIR '16, pp. 235-244, 2016.
- [14] Eriks Sneiders, "Automated FAQ Answering with Question-Specific Knowledge Representation for Web Self-Service", In Proceedings of the 2nd conference on Human System Interactions (HSI'09), pp. 295-302, 2009.
- [15] Erfan Najmi, Khayyam Hashmi, Fayez Khazalah and Zaki Malik, "Intelligent Semantic Question Answering System", In Proceedings of the IEEE International Conference on Cybernetics (CYBCONF), pp. 255-260, 2013.
- [16] Faheem Abbas, Muhammad Kamran Malik, Muhammad Umair Rashid and Rizwan Zafar, "WikiQA – A Question Answering System on Wikipedia using Freebase, DBpedia and Infobox", In Proceedings of the sixth conference on Innovative Computing Technology(INTECH2016), pp. 185-193, 2016.
- [17] Huiying Li and Feifei Xu, "Question Answering with DBpedia Based on the Dependency Parser and Entity centric Index", International Conference on Computational Intelligence and Applications, pp. 41-45, 2016.
- [18] Bolanle Ojokoh, Tobore Igbe, Ayobami Araoye, and Friday Ameh, "Question Identification and Classification on an Academic Question Answering Site", In Proceedings of the 16th ACM/IEEE-CS on Joint Conference on Digital Libraries(JCDL '16), pp. 223-224, 2016.
- [19] Hani Al Chalabi, Santosh Ray, and Khaled Shaalan, "Question Classification for Arabic Question Answering Systems", In Proceedings of the international Conference on Information and Communication Technology(ICTRC2015), pp. 310-313, 2015.
- [20] Zhao Lu, Yu-Ru Lin, Qing Zhang, and Mengwei Chen, "Classifying Questions into Fine-grained Categories using Topic Enriching", IEEE 17th International Conference on Information Reuse and Integration, pp. 166-174, 2016.
- [21] Fu Jinghong, Zhang Chunying, Wang Jing, and Tian Fang, "A Weighted Relational Classification Algorithm Based on Rough Set", I.J. Education and Management Engineering, 20-25, 2013.
- [22] Xin Li and Dan Roth, "Learning Question Classifiers: The Role of Semantic Information", Natural Language Engineering Journal, Vol. 12, pp. 229 - 249, 2006.
- [23] Waheeb Ahmed, and Babu Anto, "Classification of Arabic Questions Using Multinomial Naïve Bayes And Support Vector Machines", International Journal of Latest Trends In Engineering And Technology, pp. 82-86, special issue(SACAIM), 2016.
- [24] Waheeb Ahmed, and Babu Anto P., "Question Focus Recognition in Question Answering Systems", International Journal of Advanced Research in Computer and Communication Engineering(IJARCC), Vol. 5, pp. 25-28, 2016.
- [25] R. Al-Shalabi , G. Kanaan, J.M. Jaam, A. Hasnah and E. Hilat, "Stop-word removal algorithm for Arabic language", In Proceedings of the International Conference on Information and Communication Technologies: From Theory to Application, 2004.
- [26] Jonathan J. Webster, and Chunyu Kit, "Tokenization as The initial phase in NLP", In Proceedings of the 14th conference on Computational linguistics(COLING-92), Vol. 4, pp. 1106-1110, 1992.
- [27] Khoja Stemmer, <http://zeus.cs.pacificu.edu/shereen/research.htm>, Last Visited- December, 2016.
- [28] Hani Al-Chalabi , Santosh Ray, and Khaled Shaalan, "Semantic Based Query Expansion for Arabic Question Answering Systems", In Proceedings of the First International Conference on Arabic Computational Linguistics (ACLing), pp. 127-132, 2015.
- [29] Naushad UzZaman , Hector Llorens, and James Allen, "Evaluating Temporal Information Understanding with Temporal Question Answering", In Proceedings of the IEEE Sixth International Conference on Semantic Computing (ICSC), pp. 79-82 ,2012.
- [30] S. Harabagiu, and C. A. Bejan, "Question answering based on temporal inference.", In proceedings of the AAAI-2005 Workshop on Inference for Textual Question Answering, 2005.

- [31] Sazianti Mohd Saad, and Siti Sakira Kamarudin, "Comparative Analysis of Similarity Measures for Sentence Level Semantic Measurement of Text", In Proceedings of the IEEE International Conference on Control System, Computing and Engineering (ICCSCE), pp. 90-94 ,2013.
- [32] Issa Atoum, Ahmed Ootom and Narayanan Kulathuramaiyer, "A Comprehensive Comparative Study of Word and Sentence Similarity Measures", International Journal of Computer Applications, Vol. 135, No.1, pp. 10-17, 2016.
- [33] E. Voorhees," Overview of the TREC 2001 question answering track," In Proceedings of the 10th Text Retrieval Conference(TREC10), pp. 42–52, 2001.

Authors' Profiles



Waheeb Ahmed received his B.E. in Computer Science & Engineering from University of Aden, Yemen, 2008. He received his master's degree from Osmania University, Hyderabad, 2013, under the ICCR-CE Program. He is currently pursuing his Ph.D. in IT from Kannur University under the ICCR-GSS Program. His research interests include Natural Language Processing, Machine Learning, Artificial Intelligence, Information Retrieval and Knowledge Engineering. He has published 6 articles.



Ajusha PV received her B.Sc. in Physics, and her masters' in Computer Applications. Currently, she is doing her research in the Dept. of IT, Kannur University. Her research interests include Natural Language Processing, and Artificial Intelligence. She has published 3 articles.



Dr. Babu Anto P received his Ph.D. in Technology from Cochin University of Science & Technology, India, 1992. He is currently an associate professor and working as a research guide at the Department of Information Technology, Kannur University, Kerala, India. His research interests include Natural Language Processing, Speech Processing, Image Processing, Visual Cryptography and Artificial Intelligence. He has published 21articles, 28 conference papers and authored 3 chapters.

How to cite this paper: Waheeb Ahmed, Ajusha Dasan, Babu Anto P, "Developing an Intelligent Question Answering System", International Journal of Education and Management Engineering(IJEME), Vol.7, No.6, pp.50-61, 2017.DOI: 10.5815/ijeme.2017.06.06