World Academy of Science, Engineering and Technology
International Journal of Industrial and Manufacturing Engineering
Vol:2, No:12, 2008

Information Retrieval : Improving Question Answering Systems by Query Reformulation and Answer Validation

Mohammad Reza Kangavari, Samira Ghandchi, Manak Golpour

Abstract — Question answering (QA) aims at retrieving precise information from a large collection of documents. Most of the Question Answering systems composed of three main modules: question processing, document processing and answer processing. Question processing module plays an important role in QA systems to reformulate questions. Moreover answer processing module is an emerging topic in QA systems, where these systems are often required to rank and validate candidate answers. These techniques aiming at finding short and precise answers are often based on the semantic relations and co-occurrence keywords.

This paper discussed about a new model for question answering which improved two main modules, question processing and answer processing which both affect on the evaluation of the system operations.

There are two important components which are the bases of the question processing. First component is question classification that specifies types of question and answer. Second one is reformulation which converts the user's question into an understandable question by QA system in a specific domain. The objective of an Answer Validation task is thus to judge the correctness of an answer returned by a QA system, according to the text snippet given to support it. For validating answers we apply candidate answer filtering, candidate answer ranking and also it has a final validation section by user voting.

Also this paper described new architecture of question and answer processing modules with modeling, implementing and evaluating the system. The system differs from most question answering systems in its answer validation model. This module makes it more suitable to find exact answer.

Results show that, from total 50 asked questions, evaluation of the model, show 92% improving the decision of the system.

Keywords— Answer Processing, Answer validation, Classification, Question Answering and Query Reformulation.

Dr. Mohammad Reza Kangavari is Assistant Professor within Department of Computer Engineering(CE), Iran University and Science and Technology, Phone: +98(21)73913305; fax: +98(21)77240469, (e-mail:kangavari@iust.ac.ir)

Samira Ghandchi & Manak Golpour are with the Iran University and Science and Technology, phone:+98(21)73913305; fax: +98(21)77240469; (e-mail:samiraghandchi@comp.iust.ac.ir),(e-mail: manakgolpour@comp.iust.ac.ir).

I. Introduction

Many researches have been done in recent years on QA systems. QA systems have been expanded to answer simple questions correctly; but now researches have been focused on methods for answering complex questions truthfulness. Those methods analyze and parse complex question to multi simple questions and use existing techniques for answering them. [1]

Recent researches show that increasing the performance of system is dependent on number of probable answers in documents.

Finding the exact answer is one of the most important problems in QA systems. For this purpose, this designed model uses syntax and semantic relations together, pervious asked questions and dynamic patterns to find exact answer at least timeline.

This model work on aerology domain by forecasting the weather information based on patterns in close domain question answering system. If there is no default proper pattern in database, user can make appropriate patterns referring to English language grammar. Designed QA system just answers the questions in factoid form or one sentence.

The aim of this paper is to design and implement a new model for classification, reformulation and answer validation in a QA system. Used methodology in this system is to find correct answer in 'weather forecasting' domain with NLP techniques, syntax & semantic relation among words, Dynamic pattern and previous information about defined domain.

The main reason behind the necessity of providing the system with an answer validation component concerns the difficulty of picking up from a document the “exact answer”.

Our approach to automatic answer validation relies on discovering relations between a question and the answer candidates by mining the documents or a domain text corpus for their co-occurrence tendency[11].

In this model, first of all, questions are parsed by using semantic and syntax information in the question. Second, answer patterns based on their types are specified. Then the search engine find candidate answer document and send them to answer processing module to extract correct answers. The system filter candidate answers collection based on co-occurrence patterns and assigns a priority number to the candidate answers. Finally the system ranks the answers and sends to user for final validation in order to extract the exact answer.

Considered patterns in this program are based on English language grammar and tried to include all probable patterns. If no proper pattern find, user can make a new pattern. This paper tries to use syntax, semantic relations and existing information of pervious asked questions by users which were saved in system. Our system modeled in aerology domain but it can easily works in both close and open domain in QA systems. In Section II, we considered QA systems, section III consist of question processing part and section IV present answer processing part. Section V include the architecture of the new model and section VI discussed evaluation. Final section include conclusion of the designed model.

II. Question Answering Systems (QA)

QA is a type of information retrieval. Given a collection of documents (such as the World Wide Web or a local collection), the system should be able to retrieve answers to questions posed in natural language. QA is regarded as requiring more complex natural language processing (NLP) techniques than other types of information retrieval such as document retrieval, and it is sometimes regarded as the next step beyond search engines.[1][2]

QA research attempts to deal with a wide range of question types including: fact, list, definition, how, why, hypothetical, semantically-constrained and cross-lingual questions. Search collections vary from small local document collections to internal organization documents to complied newswire reports to the World Wide Web. QA systems are classified in two main parts [12]: open domain QA system and closed domain QA system.

Open domain question answering deals with questions about nearly everything and can only rely on general ontology [4] and world knowledge. On the other hand, these systems usually have much more data available from which to extract the answer.

Closed-domain question answering deals with questions under a specific domain (for example medicine or weather forecasting and etc) and can be seen as an easier task because NLP systems can exploit domain-specific knowledge frequently formalized in ontology.

Alternatively, open-domain might refer to a situation where unlimited type of questions are accepted, such as questions asking for descriptive [1][2]. Many searches have been done for expanding English language QA systems. Also some other works have been done on Chinese, Arabic, Spanish and … QA systems. [3]

The aim of QA systems is to find exact and correct answer for user's questions. In addition to user interaction, various QA systems contain at least three following parts:

1- Question processing
2- Document processing
3- Answer processing

III. Question Processing

As mentioned before, question, document and answer processing are three main parts of a QA system.

Important components of question processing are classification of question and reformulation.

A. Classification component

For answer extraction in a large collection of documents and texts, at first the system should know what it look for. In this case, questions should be classified regarding their types [4].

Question classification will be done before reformulation. This is for finding types of questions and answers. For this, system first should know type of question. It also helps system to omit the question in final format of answer.

Table No. 1 shows question words, type of questions and answers. Totally questions can be divided as follows:

- Questions with 'WH' question words such as what, where, when, who, whom, which, how, why and etc.
- Questions with 'modal' or 'auxiliary' verbs that their answers are Yes/No.

It is obvious that specifying type of question is not enough to find the correct answer. For example in question 'Who was the first aerologist in USA?' type of answer will be 'a person'. But if a question is asked with 'What', exact type of answer is not specified. Because the answer may be definition, number or title.[6]

For correct answer extraction, some patterns should be defined for system to find exact type of answer and then sends to document processing.[4][5]

B. Reformulation component

Question reformulation (also called surface pattern, paraphrase or answer pattern) tries to identify various ways of expressing an answer given a natural language question. This reformulation is often used in Question Answering system to retrieve answers in a large document collection. [7]

The query reformulation component converts the question into a set of keyword queries that will be sent to the search engine for parallel evaluation.

Following items are important in reformulation:

1- Use of syntax relations among words of asked question sentence.

2- Use of semantic relations among words of asked question sentence.

3- Use the existing information of pervious asked questions and answers in which a part or totally is same to user's asked question. In this case, system can use type of pervious answer for new asked question. It causes that the process of finding proper pattern and type of answer become shorter and reduces the necessary time for submitting correct answer.[8][9]

It would be possible if the system has the ability of saving information in 'Usage knowledge' database.

If all above options work together at the same time, the flexibility of system will increase. As mentioned before all flexibility of designed system in on 'Usage knowledge' part. This part is same as FAQ[1] and also can answer to new asked questions which are not totally same to Previous questions and have some differences in adverbs or verbs.

When a user asks a question, first sentence parses to its syntax components and then its keywords are selected to use in reformulation.

Table 1 Classification of question and answer

Question Classification	Sub classification	Type of Answer	Example
When		DATE	When did rain come yesterday?
Which	Which-Who	PERSON	Which person did invent the instrument of aerology?
	Which-Where	LOCATION	Which city has the min temperature?
	Which-When	DATE	Which month has max rain?
Why		REASON	Why don't we have enough rain this year?
What	What	Money / Number Definition /Title	What is the temperature of Tehran?
	What – Who	PERSON	What is the best meteorologist in Iran?
	What – When	DATE	What year do we have max rain?
	What – Where	LOCATION	What is the capital of Iran?
Who		PERSON	Who is the first meteorologist in world?

There is an important question: 'What are keywords in question sentence?'

Keywords are selected in question sentence as follow:

1- All words which are in 'quotations' and "double quotations".

2- All words that are name.

3- All words that are adverb (time, location, status).

4- All words that are main verb or modal verb.

5- All words those are subject.

6- All words that are object.

Next important subject is 'how can use keywords to make answer?'. For this propose, system uses patterns to find correct format of answer. These pattern are made regarding English language grammar.[4]

1) Rules for extract answer patterns. First step to find proper pattern is to find verb in sentence. In defined patterns verbs are totally divided in three parts:

1- Main Verb such as: 'to be' (am, is , are, was, were, been) or 'to have' (have, has, had)

2- Auxiliary verbs (do, does, did)

3- Modal verbs (can, could, shall, should, may, might, …)

Main verbs never delete in answer, but regarding type of answer its location in sentence may change. Sometimes these verbs (am, is, are …) come with another verb in 'ing' form.

But auxiliary verbs will be deleted in answers. It should be noted that 'do' may located in sentence as a main verb that can be find by semantic relations.[9][10]

If question doesn't have WH question word and question is asked with a modal verb or 'to be' then answer is yes/no.

Also if sentence doesn't have any question word (WH question or modal), system asks user that which question word make his question. After that usual process will be done.

IV. Answer Processing

Answer processing module consist of two main components: answer extraction and answer validation. First, candidate answers extract from documents which are retrieve by search engine in answer extraction module. After that we validate answers with filtering and ranking candidate answers and final system's suggested answers with user voting.

Our approach to automatic answer validation relies on discovering relations between asked question and answer candidates by mining the documents or a domain text corpus for their co-occurrence tendency[10],[11]. The underlying hypothesis is that the number of these co-occurrences can be considered a significant clue to the validity of the answer.

As a consequence, this information can be effectively used to rank the amount of candidate answers that our QA system is often required to deal with. Also we can exploit domain knowledge and answer patterns to create new answers based on co-occurrence keywords and semantic relation [5].

[1] Frequently asked questions

A. FILTERING COMPONENT

Candidate answers collection which has been sent by answer extraction feed in filtering component. These candidate collections consist of some snippets which may include the exact answer. By using answer keywords, the system finds co-occurrence words [9] and semantic relations [12] existing in database ontology and moreover related sentences from knowledge domain. By analyzing the candidate answers and using answer type and keywords, some snippets eliminate from the collection. Then the best candidate answers send for ranking.

B. RANKING COMPONENT

This component receives a list of answers which have filtered before. This list consists of the best answer from the system's point of view which is more related to the question. Ranking component classifies the answers and gives priority to them. A priority number is specified to answers by using the number of repeated answer type in the snippets and the distance of answer keywords (considering to threshold). The answer with highest priority is located at the top of the list and this task performs frequently for all answers. After that the data fetch from domain knowledge database, and the answers sent to user to validate.

C. USER VOTING (VALIDATIN)

In this step, the answers are shown to user for validation. If the top answer was the exact answer, then system would increase a validation grade in usage knowledge for [q, a] pairs. That answer will submit in database to answer next similar question. Otherwise the other candidates will be shown to user to certify. This process continuous until there aren't any other answers, then the systems asks for additional information from user and will send those information or new question to question processing module.

V. ARCHITECTURE

To increase the reliability and ability of designed QA system and to find correct and exact answer, we use dynamic pattern with semantic relations among words, verbs and keywords and also co-occurrence keywords.

In question processing module, at first the question is classified regarding linguistic theories and bases of answering questions. Then question's structure and keywords are specified by classification, send to document processing module to retrieve documents which may have proper answer.

In answer processing module, first of all candidate answers which is received from search engine, will be filtered by co-occurrence patterns and ordered based on some analyzing in system. Then the answers send to user to validate the candidates. Finally the system will present the exact answer.

A. SYSTEM COMPONETS

Designed architecture has these parts, (see fig1):

1-Question interface: In this part user writes his question by an interface. If no proper answer is given, user can write his question in another way.

2- Query Analyzer: In this part question is parsed to its particles such as subject, object, verb, noun, adjective, adverb & etc.

3- Lexicon: This part is used as vocabulary (dictionary) and contains all words that are in related domains. Also the type of word such as subject, object, verb, noun, adjective, adverb & etc is specified in this part.

4- Database Ontology: In this part questions and answers are surveyed semantically. Semantic relation among keywords saved in this database.

5-Domain knowledge: Domain information is saved as database in this part and will submit the user's answer when a web service connects to internet.

6- Question classification: Question classification is one of the important functions of most QA systems. Most researches on this subject are based on regular expression, hand writing grammar rules and other advanced techniques in natural language for question parsing and finding answers. In this part all questions are classified regarding WH question words (such as What, Where, When, Who & etc) or other question words with Yes/No answer.

7- Reformulation: In this part main question (Q) with using rules changes to a question with new format (Q'). In this part question words and punctuation which make no difference in question and answer, are deleted and the root of words will be specified. Then by the words of new question, proper patterns and information are surveyed.

8- Usage Knowledge: one of the most useful ways to find correct related answers is to use a library of the previous questions and answer. If new user's question is similar to a previous submitted question, the answer of the old question will be used as answer of new question. If the new question is different with old questions in database, new question will be sent for other steps. It should be notified, this database is new for answer validation.

9-Candidate Answers Filtering: by this part, candidate answers will be filtered based on answer type and co-occurrence patterns which were created in system. Also some answers generate dynamically based on domain knowledge and co-occurrence keywords.

10-Candidate Answers Ranking: answers rank based on distance of keywords in snippets, answer type and answer repetition. By using ranking part in our model, answer candidate collection, which be filtered, will be ranked based on their validation value.

11- User Voting: this part of the system plays the human assessment role which checks the correctness of answer and fills the validation grade in usage knowledge for the next validating which will affect on total system timeline.

12– Pattern: this is a database which is used for answer patterns and will be updated with dynamic patterns which created in system.

B. ALGORITHM

As mentioned before for each question that is written by user in natural language, some words of question are used as keywords in answer. These keywords can be used as subject, object, verb, adverbs & etc.

Designed Algorithm of this QA system is as follow:

1- User asks question through a query interface. If the question is similar to one of the previous questions which were saved in usage knowledge database, the answer of pervious question will be chosen for user's question and the system give the answer. Otherwise next step will be done.

2- Query analyzer pars question as subject, verb, object, adverb & etc. It should be noted that the type of words and synonyms of them (if is existed) were defined dynamically in Lexicon database. If system could not find the word or its type in question, system will announce and user can enter the new word and its type. In this case Lexicon database will be completed and updated.

At last a tree view result will be used in classification part. In classification part, type of question and after that type of answer will be specified.

3- The question may have a WH question word which its answer is proportionate to that question.

3-1 Asked question doesn't have any WH question word and just has a modal or auxiliary verb with Yes/No answer.

3-2 User may ask his question with a sentence that has no verb or question word such as: Temperature of Tehran.

4-After finishing these steps, for finding answer the most important part of job, query reformulation based on proportionate pattern, should be done.

5- We suppose that in document processing part, the search engine retrieves the documents in scope of the domain and based on answer patterns and important keywords.

6- Search engine send candidate answers collection to answer processing module. Answer extraction part will extract candidate answers from retrieved documents. Then these candidate answers pass to filtering unit.

7- Based on co-occurrence words and semantic relations existing in database ontology, answer type and keywords which extracting in question processing module, system filter candidate answers collection. Therefore some answers which are not related with the asked question will be eliminated.

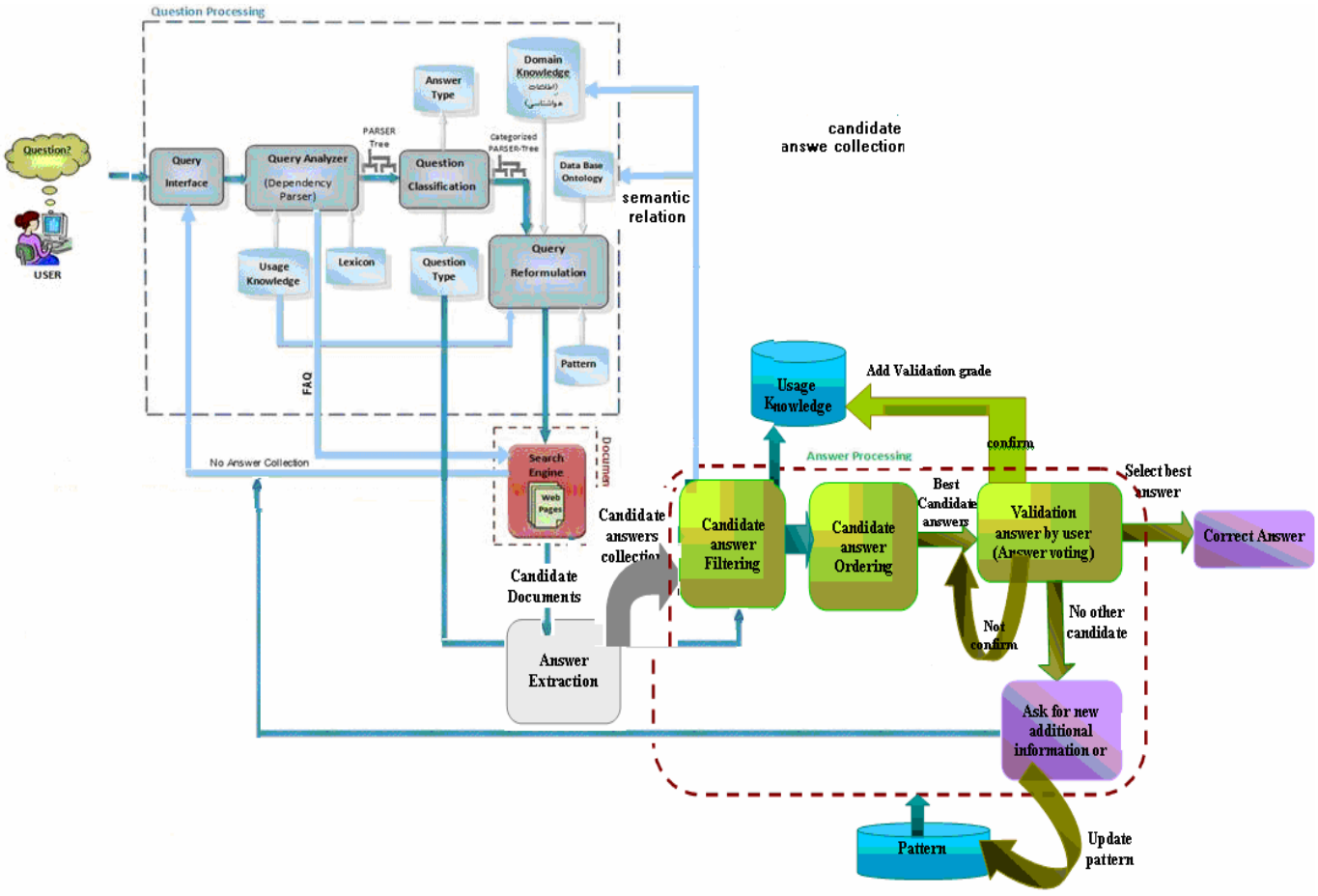

Fig. 1 Question answering system architecture

8- The remained answers rank by keywords distance and the frequently rate of answer keywords in snippets. In this case the filtered answers obtain priority and located in an ordered list.

9- The answers with high priority show to user for validation. Then the answers get a validation answer grade and save it in usage knowledge. If the user accepts the suggested answer which system presented as an exact answer, the algorithm will be finished.

10- If not, the algorithm send next set of candidate answers with priority to user from the list. This task performs recursively.

11- Finally if the user doesn't accept any answer from the candidate list, the system asks for new question and request for additional information from the user and sends them to question processing module. Also update the pattern database to eliminate non efficient patterns.

VI. Process of Model

For increasing efficiency and finding exact answer, system uses data base ontology. This database include co-occurrence words such as rain and umbrella [6] and also semantic words which are near in meaning, such as 'temperature' and 'degree' in weather forecasting domain.

Flexibility of designed system is based on usage knowledge. It means that if the new question is totally or nearly similar to a question which was asked before, system can use the generated answer. Also we use a field named answer validation grade to decrease system's useful timeline.

If an answer is valid the system automatically adds one number to this grade. Then if user ask repeated question, the answer with highest grade will be select as valid answer.

New asked question is parsed in query analyzer part to its components. Then all of these components check with data in usage knowledge to find the probable similarity with pervious questions. If during checking, the structure of asked question totally is same as data in usage knowledge, certainly the answer of new asked question is same to answer of pervious question. But if some differences find between new asked question and data (such as question word, proposition, adjective, name and adverb) then system uses 'word ontology' to find synonyms of different parsed words to find similarity between new asked and pervious questions. At last if there was any answer for synonyms word in previous question, system uses this answer for the answer of new asked question.

For example during checking, if two words such as 'temperature' in new asked question with 'degree' in pervious questions is different and in 'word ontology' these two words were saved as 'synonyms', also a previous question with 'degree' was asked, then system takes these two questions and the type of their answers, same even if they have different adverbs.

It should be noted that different adverbs in two same questions, have no effect in type of question. This option is important for question words that have more than one type of answer (such as 'what' that its answer type may be 'number', 'title' or "definition').

If the structure of question has totally different, it means no similar question exists in usage knowledge, system uses other defined patterns to finding answer of question.

Suggested model by using co-occurrence technique can increase the validity of the candidate answers and also exploiting from validation value caused the efficiency of system affected. In addition, in final step user can check the validity of answers and select the best validated answer/answers. System will appropriate a validation value for selected answer and store that value in usage knowledge. Likewise, system receives a score between 0-100 from user which shows user satisfactions from system operation. This score will use for evaluation of the system. Because of this measure for evaluation which exploit user viewpoint, the percent of validation is so high.

If there are some words which are co-occurrence in candidate answer sentence, there is strong probability that the answer will be valid. The ontology for the weather events consists of event concepts, which are similar to Synset in WORDNET [15]. For example, rain and umbrella are in same event concept in the domain ontology for weather events, because the questions about using umbrella are usually asking about raining (e.g. Will I need to bring umbrella tomorrow? and Will it be raining tomorrow?)

VII. Evaluation

The model implemented base on dynamic patterns, syntax and semantic relation among words, co-occurrence keywords, answer validation value, and use previous information (question and answers) that saved in usage knowledge.

Chart No.1 shows type and quantity of questions (such as questions which ask for "quality/status, quantity/amount, location, time/date, person and defined/descriptive questions").

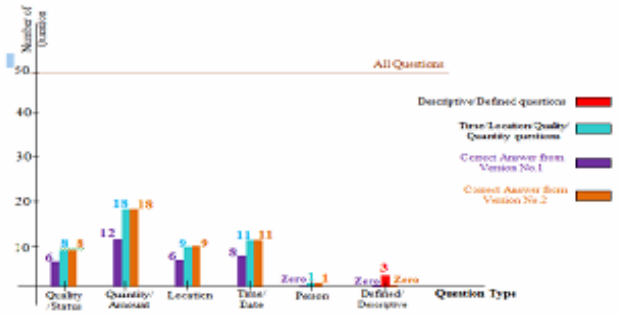

Chart No.1- Comparison of Question Types and Correct Answers

The model is capable to control words of question semantically and also use co-occurrence word relations for validate the answer. Other advantage of this model is the ability of defining new answers by system from domain knowledge, keywords and answer patterns. Also the model used an answer validation value which affect on system's

timeline. Because our domain is restricted on aerology, and if frequent question increase in system, number of the validated answer also increase. This means that the validity of answers will be increased.

Moreover, model works with dynamic patterns. Also this model of program is capable to answer a sentence with more than one question word individually or a sentence without question word or a multi sentence text that has a question.

For evaluation of the implemented system, 50 questions were asked by 20 various persons in age and knowledge in different location and time situations.

From total 50 asked questions, evaluation of the model, show 92% improving the decision of the system.

Chart No.2 shows evaluation if the model base on user voting.

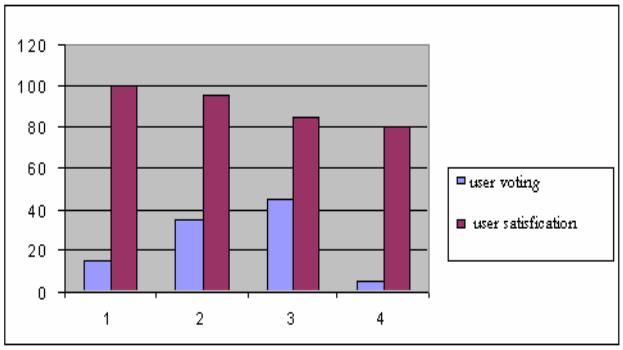

Chart No2. Evaluation

As before noted, this model improve the precision of question answering systems by working on query reformulation and answer validation modules. The model exploit syntax and semantic relation for making dynamic patterns to reformulate asked question and also develop answer validation part by using words co-occurrence techniques and apply user assessments to validate answers. Then system's performance increased.

Finally model use usage knowledge to decline total timeline for answering to question.

VIII. Conclusions and Future Works

RDQA, working on small document collections and restricted subjects, seems to be a task no less difficult than open-domain QA. Due to candidate scarcity, the precision performance of a RDQA system, and in particular that of its IR module, becomes a problematic issue. It affects seriously the entire success of the system, because if most of the retrieved candidates are incorrect, it is meaningless to apply further techniques of QA to refine the answers.

The simplest approach to improve the accuracy of a question answering system might be restricting the domain it covered. By restricting the question domain, the size of candidate answer collection becomes smaller.

Reformulation is an important part for understanding the interplay of information retrieval. There are three steps in patterns of reformulation by user: format, contents and source.

The main goal of rewriting question is asking question in another new format by user in which less time and sources are used for search. Reformulation of questions is one of the most difficult parts in user's domain, even in web that seems learning and using documents is simple. Understanding the behavior of question and designed software for supporting of these behaviors are important problems in this part as the QA systems based on reformulation.

So for increasing the performance and getting exact correct answer, this component should use syntax and semantic relations together at same time and also use the existing information of pervious asked questions by users which were saved in program.

One of the important specifications of reformulation component is the ability of being used separately in other QA systems. It is for, not having any relation to a special domain and can be used in open and closed domains.

The designed system considers following step to increase its proficiency:

- Semantic analysis
- Syntax analysis
- Dynamic pattern
- Use the pervious data(usage knowledge)
- Using questions with related subjects,
- Adding a web services as a rich source in domain ontology

In addition the question processing module, the improvement of answer processing module will be complete the question processing task in efficiency of the QA system, because the system must return correct answer. Then by improving the answer processing module the system can able to present exact answers, because it perform on close domain and deal with frequent questions and use validations patterns. Another reason is that the exact answers obtained by filtering candidate answers collection which perform in several steps, therefore the answers select from restricted collection and this makes the algorithm more efficient. This model by using a validation grade is more effective than the other models in total response time. This grade is null at the beginning of the system but by using QA system this field will increase and affect on response access time.

Future researches should consider factors that lead users to reformulate their questions. Also new research should be done to gather more information in various levels of understanding, effectiveness and situations. Methods of gathering multi information such as documents, interviews, reports and etc. should be done. In addition that we must improve the answer processing module by identification new kind of patterns and try to decline the timeline to find the exact answer which is performed here by using validation grade and usage knowledge database.

REFERENCES

[1] Demner-Fushman, Dina, "*Complex Question Answering Based on Semantic Domain Model of Clinical Medicine*", OCLC's Experimental Thesis Catalog, College Park, Md.: University of Maryland (United States), 2006.

[2] Doan-Nguyen Hai, Leila Kosseim, "*The Problem of Precision in Restricted-Domain Question Answering. Some Proposed Methods of Improvement*", In Proceedings of the ACL 2004 Workshop on Question Answering in Restricted Domains, Barcelona, Spain, Publisher of Association for Computational Linguistics, July 2004, PP.8-15.

[3] Green, W., Chomky, C., Laugherty, K. BASEBALL: "*An automatic question answer".* Proceeding of the western Joint Computer Conference, 1961, PP. 219-224.

[4] Figueira, H. Martins, A. Mendes, A. Mendes, P. Pinto, C. Vidal, D ,"*Priberam's Question Answering System in a Cross-Language Environment*",LECTURE NOTES IN COMPUTER SCIENCE, Volume 4730, 2007,PP. 300-309.

[5] Dan Moldovan, Sanda Harabagiu, Marius Pasca, Roxana Girgu, " *The Structure and Performance of an Open-domain Question Answering System*", Proceedings of the 38th Annual Meeting on Association for Computational Linguistics Hon Kong, 2000, PP. 563-570,.

[6] Cody Kwok, Oren Etzioni, Daniel S. Weld, "*Scaling Question Answering to the Web*", Proceedings of the 10th international conference on World Wide Web, Hong Kong , 2001,PP. 150-161.

[7] Maria Varges, Verona and Enrico Motta, "*AQUA, A Knowledge-Based Architecture for a Question Answering System*", Tech Report Kmi-o4-15, Knowledge media institute Milton Keynes, England, 2004.

[8] Lehnert, W. G. "*A conceptual theory of question answering*". In International Joint Conference on Artificial Intelligence (IJCAI 1977), 1977, PP. 158–164.

[9] Garg, A. X.; Adhikari, N. K. J.; McDonald, H.; Rosas-Arellano, M. P.; Devereaux,P. J.; Beyene, J.; Sam, J.; and Haynes, R. B. E.ects of "*computerized clinical decision support systems on practitioner performance and patient outcomes*". The Journal of the American Medical Association 293(10), 2005, pp.1223–1238.

[10] Alexander Panossian, Georg Wikman , "*Knowledge Bases in Medicine: a review*". Journal of Ethno pharmacology, Bulletin of the Medical Library Association ,Volume 118, Issue 2, 23 July 2007, PP. 183-212.

[11] Magnini, B., Negri, M., Prevete, R., Tanev, H.: "*Comparing Statistical and Content-Based Techniques for Answer Validation on the Web*", Proceedings of the VIII Convegno AI*IA, Siena, Italy, 2002.

[12] Magnini, B., Negri, M., Prevete, R., Tanev, H.: "*Is It the Right Answer? Exploiting Web Redundancy for Answer Validation*", Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL-2002), Philadelphia, PA. 2002.

[13] Magnini, B., Negri, M., Prevete, R., Tanev, H.: "*A WordNet-Based Approach to Named Entities Recognition*", Proceedings of SemaNet02, COLING Workshop on Building and Using Semantic Networks, Taipei, Taiwan, 2002.

[14] Hai Doan-Nguyen, Leila Kosseim: "*Improving the Precision of a Closed-Domain Question-Answering System with Semantic Information*", ACL 2004 Workshop on Question Answering in Restricted Domain,2004-acl.ldc.upenn.edu

[15] Fellbaum, "*WordNet: an Electronic Lexical Database*". The MIT Press , 1998