I.J.Modern Education and Computer Science, 2012, 3, 28-34
Published Online April 2012 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijmecs.2012.03.04

AISQA - An Artificial Immune Question Answering System

Mohsen Shakiba Fakhr
Department of Computer Engineering, dezful branch, Islamic Azad University, Dezful, Iran
Email: Shakibafakhr@yahoo.com

Mohammad Saniee Abadeh
Electrical and Computer Engineering College, Tarbiat Modares University, Tehran, Iran
Email: Saniee@modares.ac.ir

***Abstract*—** Question answering (QA) is the task of automatically answering a question posed in natural language. At this time, there exists several QA approaches, and, according to recent evaluation results, most of them are complementary. Some of them use the evolutionary algorithms, such as the genetic algorithm, in itself. In this paper we propose a question answering system that uses the artificial immune algorithms, for searching in the knowledge base to find the right answer. This algorithm is one of the evolutionary algorithms. Search is based on two features: (i) the compatibility between question and answer types, (ii) the overlap and non-overlap information between the question-answer pair. Experimental results are encouraging; they indicate significant increases in the accuracy of proposed system, in comparison with the previous systems.

***Index Terms*—** *QA, GA, Artificial Immune System, Mutation.*

I. Introduction

Information Retrieval (IR) Systems, receive several keywords from the user, and the search engine retrieves all the related documents from its document repository in a limited time. Most of retrieved documents are just syntactically –and not semantically- related to the user query. These engines receive the users query that consist of several keywords, and instead of giving exact answers to the users question just retrieve the documents that are relevant to users query [1].

These systems have some major problems. First, those users have question, but instead the question some keywords should be entered. On the other hand, usually users have a problem to convert the question to the appropriate keywords, and this conversion requires skill, that must be achieved over time. In addition, several keywords cannot make the user intention, that this issue, sometimes impossible to make this conversion. So we can say that the use of keywords, not proper and thorough method for communication between the system and user. On the other hand, usually, users are looking for exact answers, while the output of these systems is a great document that may not have the correct answer in itself. Thus, user is forced to read a large number of documents, to find their desired answers.

Users need exact and accurate information and don't like to waste their time by reading all retrieved documents to find the answer, and IR systems are not sufficient for this reason [2]. So, a new kind of IR, named Question Answering (QA) systems appeared from the late 1970's and early 1980's. In these systems, the user ask his/her natural language question with no restriction in its syntax or semantic. The system is responsible for finding the exact, short, and complete answer at the shortest possible time. To do this, a QA

system applies both IR and NLP techniques [3]. In a subdivision, question answering systems are divided in two categories [4]:

- **Restricted domain**. Responds to questions on a particular domain (eg medical or car maintenance), and can be use of specific knowledge of its domain, to natural language processing.
- **Open domain**. That almost dealing with any question, and can rely on the global ontology and public knowledge.

Another division for QA systems is based on the number of languages accepted by these systems. Monolingual systems, receive the question, and respond to it, only with one language. Another group that is called multilingual systems, have ability for understand and respond to questions that include several different languages [5].

This paper will present briefly review on the question answering system, and related work in this area. The third section is dedicated to the proposed work, in the fourth section, the proposed system performance is evaluated and compared with genetic algorithm, and the fifth section of paper is concluded.

II. Related Works

QA systems which are based on searching among a set of documents are usually composed of three main modules [6]: (1) question analysis and extension (2) document retrieval (3) answer extraction. The first module analyses the user question to extract the type of question and the expected type of answer [7] or extends it to be used by the next modules. The second module relates to retrieving relevant documents to the user query. It can be replaced by a search engine. The third module extracts the final answer from the documents retrieved by the second module.

All question answering systems, have three steps above, but different methods are used to implement this process.

The first devices for access to the information, were textual information retrieval systems, that despite the simple, are useful, and very widely used. An example of this systems, are Google, Altavista and MSN Search, where used to find relevant documents on the Internet. Some of information retrieval systems, are designed for use in textual collection, out of the Internet, such as the SMART [8] and PRISE [9].

Web Question Answering System, is another example of question answering systems, which used the genetic algorithm for ranking. In this system, at first, words will be sent to the Web, and sentences that include the answer, are retrieved. Set of retrieved sentences, are matched with known previous paragraphs, so new answers to be extracted. Therefore, matching procedure, is an important parameter. Two strategies based on genetic algorithm, is proposed to improve the matching:

a) *GASCA*, Trained of Syntactic patterns, derived of (sentence, answers) pairs. For matching, and find alignment, block of words to be translated into near blocks of zeros and ones. Then, according to their fitness, new blocks to be made by the "mutation" and "crossover" operators, that make more probability of matching between training patterns with query.

b) *PreGA*, Use the semantic relation, to matching the query and training patterns. The previous strategy is based mainly on syntactical patterns, then, if there is not enough syntactical evidence to properly align contextual patterns, the answer will not be unambiguously identified. But with using of this strategy, improved matching between the educational patterns and query [10].

The basic algorithm that used in this system, is shown in Figure 1.


```
Algorithm GA_QA
input: num_iter, pop_size, Q
begin
    Rnd[1] ← create initial population(1,pop_size);
    Evaluate population (rnd[1]);
    Store ( (maxfit), loc(maxfit), db(maxfit));
    for  i=1: Num_iter
        CAC ←  Crossover (rnd[i],pc);
        MAC ← Mutate (rnd[i], pm);
        Rnd[i+1]←selectPopulation(rnd[i],CAC,
        MAC);
        Evaluate population (rnd[i+1]);
        Store  ((maxfit), loc(maxfit), db(maxfit));
    end
Return db(max (max_fit));
end
```

Figure 1. GA_QA Algorithm

Num_iter: is the number of implementing stages mutation, pop_size: is the number of initial population size, Q: is the user question, Rnd: is an array of initial population, Maxfit: is the sentence with maximum fitness, CAC: is the result of Crossover and MAC: is the result of Mutation.

In this paper, reviews the Question Answering System, called *AIS_QA*. This system can answer the questions about the inventors. The system knowledge base, is made of the web pages, so that a web page is given as input to the system, then separated parts of it, stores as a sorted knowledge base. Then, using an artificial immune algorithm, select the best sentence available in knowledge base as the answer. The advantage of our proposed system in comparison with Question Answering Systems that uses genetic algorithms to search in the knowledge base, is high accuracy to answering the questions.

III. System Architecture

This section introduces the structure and function of *AIS_QA* System. This system has implemented, as a restricted domain Question Answering System. Such as most similar systems, this system is composed of three main components too. The main components of this system are: "sentence analysis system", " retrieval and extraction answer system" and " Ranking System". The overall structure of system can be seen in Figure 2.

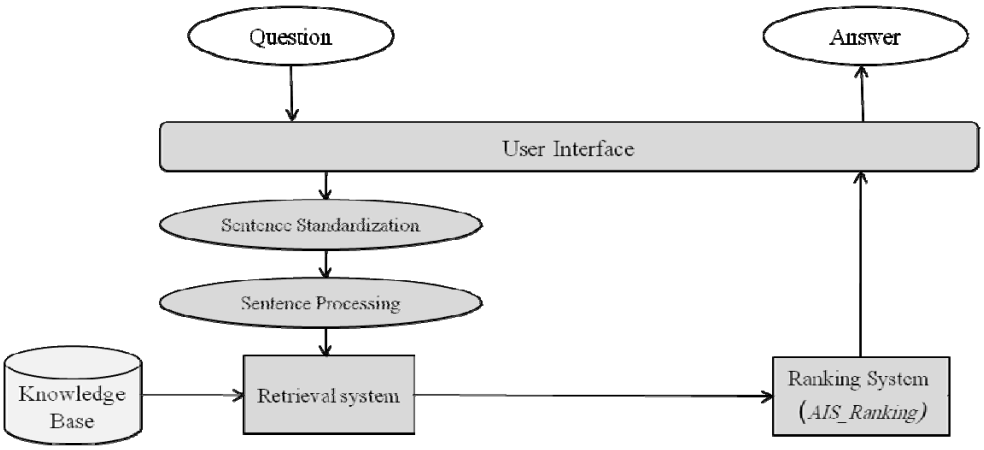

Figure 2. AIS_QA System

The ranking systems *(AIS_Ranking)*, is composed of several components. Figure 3 shows this system. In this section, by using an artificial immune system, candidate sentences are ranked, and the highest rank, is placed in the output, as an answer.

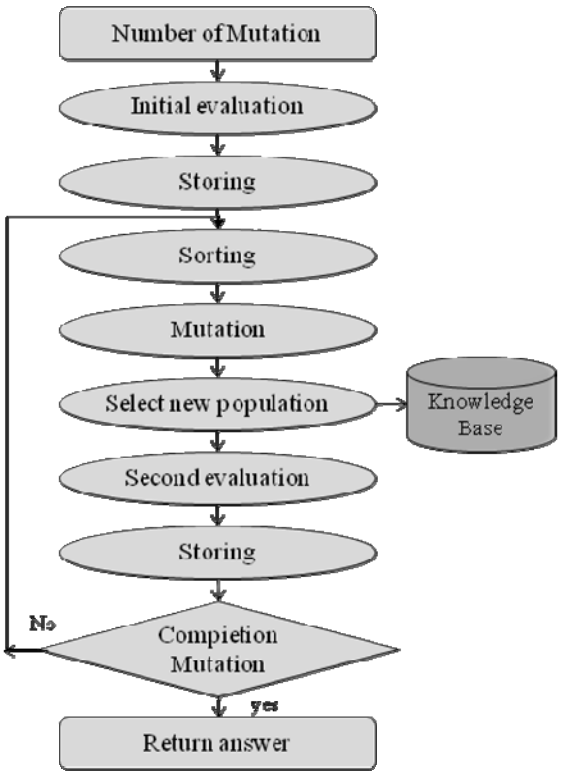

Figure 3. Ranking system (AIS_Ranking)

A. User Interface

In this section, the system receives question in natural language, along with *pop_size* and *Num_iter*.

Num_iter, is the number of mutation. Whatever this number is greater, hence the number of mutations to be great, and thus, is more probability to find answers. But, if this number was too high, reduce the speed of program implementation.

pop_size, is the initial population size. Considering that, the selected population, are chosen in the quad categories for mutations, this number must be multiple of four. Otherwise, by a phrase, becomes to coefficient of four. If the initial population size, was much more, the probability of finding the best sentence, before the mutations, is added. But if its value becomes too high, reduce the speed of program implementation.

B. Sentence Standardization

In this section, the user question is inserted, from the previous stage. Then standardization function applies on it. Standardization is three stages.

First step: check all the words in question, and the words that written in capital letters, are converted to the lowercase.

Second step: all extra words (eg. am, and, or, if, is, a, as, an, to, for, the) are removed from the question.

Third stage: all the words in question, are examined to find the words that terminated with the ('s, es, er, ional, ion, ors, ive, ions, ed, or, ing), and removed these, from word endings.

C. Sentence Processing

In this section, the previous stage output that contains the question keywords, is processed, to detected questions type, so, based on question type, can also predicted the type of answer.

D. Retrieval system

In this section, for initial evaluation, some of sentences in the knowledge base are randomly extracted. Evaluation of selected sentences will be discussed in the ranking system.

E. Ranking System

The selected sentences, which extracted from the knowledge base, are ranking. Candidate sentences ranking, is based on question keywords matching with selected sentences and questions types matching with selected sentences type. This system is composed of five stages: Initial evaluation, Sorting, Mutation, Second evaluation, Return answer.

- Initial evaluation: At this point, two fitness for each sentence in knowledge base, are used «*fit1*, *fit2*». *fit1*, is the matching result of all words in question, with any words in all knowledge base, and *fit2*, is for reviews the question type matching with the sentences type of knowledge base. For example, for given question, the corresponding response is as follows:

  Who `invented the radio?`

  `The radio was invented` ***by*** `Nikola Tesla.`

  As can be seen, in response to the "***who***" question type, the "***by***" word, exists. So, if the question type, is "***who***", then, the answer, must include "***by***" word and thus *fit2*= 1. Similarly,

we can determine the type of other questions and get the type of expected answer.
After calculating these value for all selected sentences, total fitness for i'th sentence in the knowledge base, is calculated by equation 1.

$$Global\ fitness(i) = (Fit1(i)*W1) + (Fit2(i)*W2) \quad (1)$$

W1 and *W2*, are respectively the coefficient of *fit1* and *fit2*, and they values are respectively 0.4 and 0.6.

- Sorting: This step includes the sorting *Global fitness* and *initial population* array, with ascending order. Sorting is done by the bubble sort.

- Mutation: The mutation operation is done by artificial immune algorithm. If the sentences, have the fitness be over 1.1, one step mutation, if the fitness is between 0.4 to 1.0, four step mutations, and if the fitness is between 0 to 0.3, we have six steps mutations. Here, the "Backward Mutations" and "Random Mutations" is used. In Figure 4, the proposed algorithm, named as *AIS_QA* is presented, which used to improve the initial population.

- Second evaluation: On the new population, which generated from the previous stage, the evaluation will be done, same as the first stage. Then, all the above steps, from sorting step, are repeated in *num_iter* times.

- Return answer: After the mutation was performed on the population in *num_iter* times, a sentence that has the highest fitness value, is referred as the output of ranking system.

Algorithm AIS_QA

```
input: num_iter, pop_size, Q
begin
    Rnd[1] ← create initial population(1,pop_size * 10);
    Evaluate population (rnd[1]);
    Store ( (maxfit), loc(maxfit), db(maxfit));
    for  i=1: Num_iter
        Sort (fit[i], rnd[i]);
        MAC ← Mutate (rnd[i], pm);
        Rnd[i+1] ← selectPopulation(rnd[i], MAC);
        Evaluate population (rnd[i+1]);
        Store  ((maxfit), loc(maxfit), db(maxfit));
    end
Return db(max (max_fit));
end
```

Figure 4. AIS_QA Algorithm

Num_iter: is the number of implementing stages mutation, pop_size: is the number of initial population size, Q: is the user question, Rnd: is an array of initial population, Maxfit: is the sentence with maximum fitness and MAC: is the result of Mutation.

F. Display the final answer

This section is used to prevent display the incorrect answer. Note that, sentences that are not contain the correct answer, can have the fitness equal to one, will prevent from display them in output. So that, if the highest fitness, is less than 1.1, <NOT FIND> message is written in the output. Otherwise, one sentence that has the highest fitness is displayed as the output in the user interface.

IV. EXPERIMENTAL RESULTS

This section examines the results of different experiments, which performed on the proposed method, faced with variety data. Also, the proposed artificial immune algorithm, are compared to genetic algorithm.

First, user question and initial population size and the number of mutation operators, entered into system, then, candidate sentences are scoring, and finally, a sentence with highest score, are displayed as response to the user. In this paper, two scoring methods that use the genetic algorithm and artificial immune algorithm, are compared with each other.

In Figure 5, a sample implementation of *AIS_QA* system, with initial population equal to 4 and number of implementing mutation, equal to 10, is displayed. Moreover, In Figure 6, the fitness of this system is displayed.

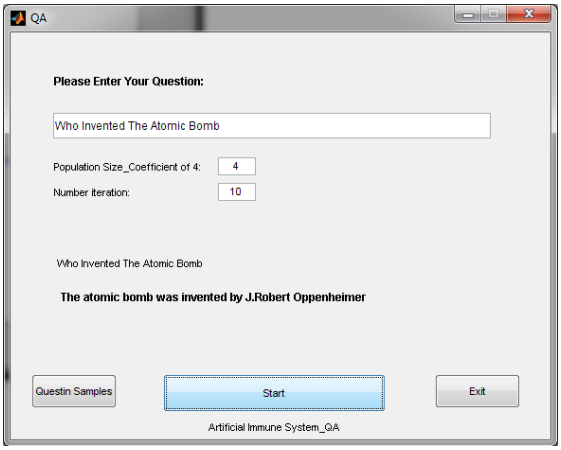

Figure 5. AIS_QA System

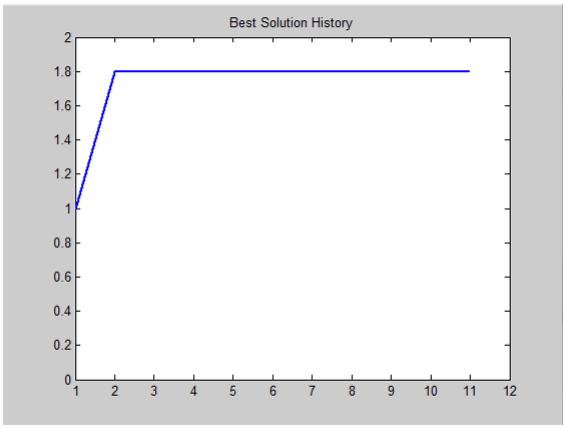

Figure 6. Fitness of AIS_QA System

As can be seen in Figure 6, in the first stage, the fitness of system is equal to 1, then, after one stage of mutation, the fitness equal to 1.8. Considering that this value is greater than 1.1, So, the system is found the answer.

In Figure 7, the fitness percentage of *GA_QA* and *AIS_QA* system, with the initial population equal to 8 and implementing stages 1 to 10, are displayed.

Figure 7.Compare the fitness percentage of GA_QA and AIS_QA system,with the initial population equal to 8

In Figure 7, in tenth stage, the fitness average of *AIS_QA* system is 100%, while *GA_QA* system is 24%.

In Figure 8, the fitness percentage of *GA_QA* and *AIS_QA* system, with the initial population equal to 12 and implementing stages 1 to 10, are displayed.

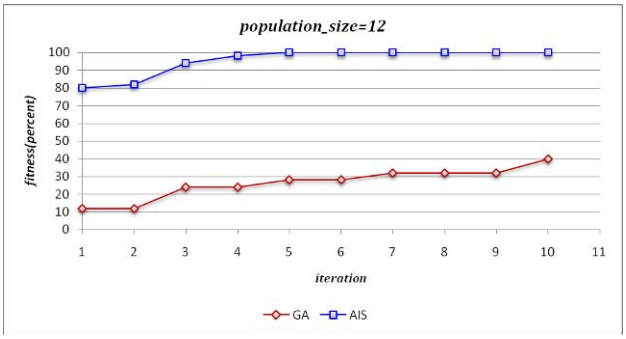

Figure 8. Compare the fitness percentage of *GA_QA* and *AIS_QA* system, with the initial population equal to 12

In Figure 8, in fifth stage, the fitness average of *AIS_QA* system, and thus the accuracy of this system is 100%, that 72% improved, in comparison with the *GA_QA* fitness.

In Figure 9, fitness percent average of *GA_QA* and *AIS_QA* system, with implementing stages 1 to 10, are displayed.

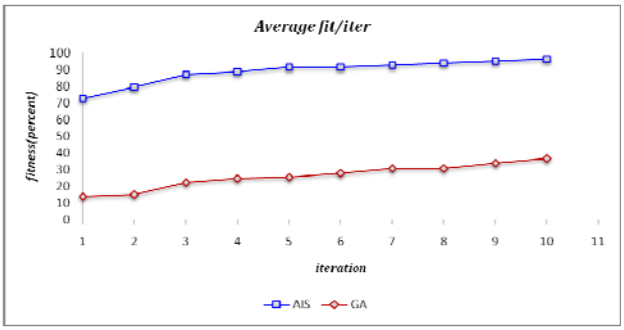

Figure 9. Compare the fitness percentage average of GA_QA and AIS_QA system, than implementing stages mutation

In this Figure we see the fitness percentage of the proposed system, in all stages is higher than the previous system.

V. Conclusions

In this paper, we presented a restricted domain question answering system based on the knowledge base, which uses the artificial immune algorithm for ranking.

The system knowledge base, is composed of the structured texts. To making this knowledge base, used the non-structured web pages. Processors and standardized components of this system, converts the natural language question to the keywords. By using this keywords, are scoring the sentences in the knowledge base. Scoring is based on the question keywords matching, with the sentences in the knowledge base, and also, the question type matching with the knowledge base sentences type. After scoring the sentences in the knowledge base, the highest ranking sentence will be displayed in the user interface. According to evaluations, the overall average accuracy of proposed system, have been significantly improved in comparison with *GA_QA*.

REFERENCES

[1] M. Shamsfard, M. Arab Yarmohammadi, " A Semantic Approach to Extract the Final Answer in SBUQA Question Answering System", International Journal of Digital Content Technology and its Applications, Volume 4, Number 7, 2010.

[2] X. Li, D. Roth, "Learning question classifiers", In COLING 2002, The 19th International Conference on Computational Linguistics, pp. 556–562, 2002.

[3] A. Ghobadi-Tapeh, M. Rahgozar, "A knowledge-based question answering system for B2C eCommerce", Elsevier Knowledge-Based Systems 21-journal homepage: www.elsevier.com/locate/knosys, PP. 946- 950, 2008.

[4] http://en.wikipedia.org/wiki/Question_answerig

[5] H. Baayen, et al., "Advances in Open Domain Question Answering", Published by Springer, P.O. Box 17, 3300 AA Dordrecht, The Netherlands, 2008.

[6] M.A. Yarmohamadi, "Answer Extraction from retrieved documents in a question answering system", MS Thesis, Computer Engineering Department, Faculty of Electrical and Computer Engineering, Shahid Beheshti University, Tehran, Iran, 2007.

[7] X. Lin, H. Liu, P. Lin, M. Wang, "Chinese Question Classification Using Alternating and Iterative One-against-One Algorithm" , Journal of convergence information technology, Volume 5, Number 3, May 2010.

[8] G. Salton, "The SMART Information Retrieval System", Prentice Hall, Englewood Cliffs, NJ, 1971.

[9] D. Dimmick, G. O'Brien, P. Over and W. Rogers, "Guide to Z39.50/Prise 2.0: Its Installation, Use, & Modification", Gaithersburg, Maryland, USA, 1998.

[10] A.G. Figueroa and G. Neumann,"Genetic Algorithms For Data-Driven Web Question Answering", journal, Evolutionary Computation, Volume 16 Issue 1, MIT Press Cambridge, 2008.

Mohsen Shakiba Fakhr is a lecturer at the Shoushtar University; He received his M.Sc. degree in Computer Engineering (Computer Architecture) from Islamic Azad University, Iran, in 2012. He received his B.Sc. in Computer Engineering (Hardware) from Islamic Azad University, Iran, in 2008. His main research interests concern Artificial Intelligence, Question Answering Systems and Natural Language Processing.

Mohammad Saniee Abadeh is an assistant professor at Electrical and Computer Engineering of Tarbiat Modares University, Tehran, Iran. He received his Ph.D. degree in Computer Engineering (Artificial Intelligence) from Sharif University of Technology, Iran, in 2008. He received his M.Sc. degree in Computer Engineering (Artificial Intelligence) from Iran University of Science and Technology, Iran, in 2003. He received his B.Sc. in Computer Engineering (Software) from Isfahan University of Technology, Iran, in 2001. His main research interests concern Artificial Intelligence and Robotics.

