The Design and Implementation of an Interactive Intelligent Chinese Question Answering System

Faguo Zhou Bingru Yang

School of Information Engineering, University of Science and Technology Beijing, Beijing 100086, P. R. China

Abstract

At first, this paper reviews the concept and development history of the Question Answering system and several common types of Question Answering systems are briefly introduced. Then concerning the Question Answering system based on knowledge-base, a theoretical model of an interactive intelligent Question Answering system (IICQAS), technology and the main algorithms concerned are provided. In the end, this paper gives the prospects of development for this kind of intelligent system.

Keywords: Chinese question answering system, Frequently asked questions, Knowledge-base, Sentence similarity

1. Introduction

Question-Answering (QA) system[1], also referred to as Human-Machine Conversation(HMC) system, is a machine system by which users can input questions in natural language and a concise, accurate and user-friendly answer, usually in the form of a short text, will be given. QA system is a hot topic of research at present. It can allow users to ask questions in natural language and give a concise and accurate answer instead of some relative web pages. Therefore, compared with the traditional search engines based on key-word matching, QA system can better satisfy the needs of users or retrieval and find the answer the users need accurately in a quicker, more convenient and more effective way.

But, the present QA systems mainly offer simple mechanical answers. They are often criticized for being not user-friendly, unintelligent and uninteractive with users. There is a lot of space for them to be improved.

Considering the above-mentioned problems, an interactive intelligent Chinese QA system based on tree model is proposed in this paper.

2. The Theoretical Model and main Technology of IICQAS

2.1. The Origin of QA system and its main implementation techniques

The research of QA system originated in the 1960s. In the reference [2], we can find the memory organization, knowledge representation, the theorem-proving of the deduction mechanism and applications of Intelligent QA systems. A preliminary introduction of the system is also given in that reference. In the past several years, both in China and abroad, the research in QA system has achieved rapid progress and brought a lot of research results.

The research of Chinese QA system originated in the late 20th century and has achieved a lot in the past ten years. Many experts have done a lot in this field and have brought many beneficial results, such as Chinese QA system based on ontology [3], semantic similarity [4], data-mining [5], semantic web [6] and retrieval [7], etc.

The intelligent Chinese QA system based on knowledge base is the focus of research. The present QA systems mainly offer simple mechanical answers. They are often criticized for being not user-friendly, unintelligent and uninteractive with users. There is a lot of space for them to be improved.

Considering the above-mentioned problems, an interactive intelligent Chinese QA system based on tree model is proposed in this paper.

2.2. The Commonly Seen QA Systems

The commonly seen QA systems [8] are as follows:

(1)ChatBot[9]: It is a QA system that imitates the language habits of human beings and the answers provided are user-friendly. In ChatBot, complicated algorithms are not concerned, and in fact, it mainly

applies pattern-matching methods to search the most suitable answers. The common systems of ChatBot abroad are ALICE, Cyber Ivar, etc. In China, the representative system of ChatBot is Xiao-i.

(2)QA system based on knowledge-base: It is a system that is based on one or several knowledge bases and applies the techniques of retrieval, deduction, etc. to understand and answer users' questions. The most common one is the QA system based on Frequently-Asked Questions (FAQ) [10]. The algorithm of the question similarity computing is the main technology. The present systems of this kind are mainly at the beginning level. They can only answer questions mechanically and lack intelligence and interactivity.

(3) Retrieval-based QA system[11]: In response to users' questions in natural language, this kind of system finds relative texts or web pages from WWW and gives them to the users. If there is no result, the information concerning the questions will be submitted to the system manager to solve. In China, the main retrieval-based QA systems are Baidu Zhidao, iAsk, Xiao I, etc.

2.3. The Solution Program and Theoretical Model of IICQAS

To tackle the single, flat system architecture of the Chinese QA system based on knowledge-base, a multi-layer, interactive solution program is proposed in this paper.

The question that a user proposes may not be a simple question and the answer to it may not be stored in the knowledge-base. It is also possible that the question is caused by several factors and the question can be expressed in several ways. All these have been stored in the knowledge-base with relevant records. And the factors that have caused the question may be caused by several other factors. Some way to express the question may also be expressed by some other ways. There are multi-layers to this situation.

Therefore, a tree model is proposed in this paper to solve this problem. If the maximal number of the factors that have caused a question is m, the tree is M-ary Tree.

2.4. The main Techniques and Algorithms

The main techniques and algorithms in the interactive Chinese QA system in this paper are as follows:

(1)Tree creation and visit algorithm

(2)Semantic interpretation and the algorithm for question similarity computing

(3)Self-learning of users' visit path

(4)The building and maintenance of knowledge-base

A detailed explanation of the above is as follows:

3. The Data Structure and Visit Algorithm of the Tree Model

3.1. The Tree Model of IICQAS

The tree model of the interactive intelligent Chinese QA system is shown as Fig. 1.

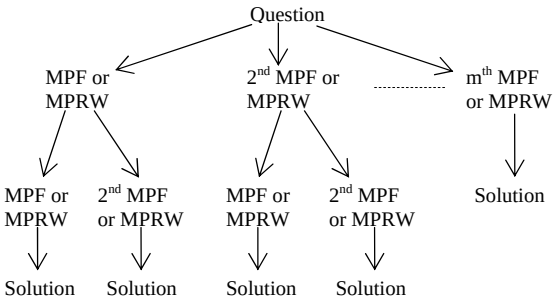

Fig. 1: Theoretical Model of the Interactive Intelligent Chinese Question-Answering System

In this figure, MPF is abbreviated of most possible factor, and MPRW is abbreviated of most possible representation way.

The root node matches the question that users have input. The leaf node is the solution to the question that is caused by some factor.

When the user has input a question, by means of semantic interpretation the system will match the most similar question from the knowledge-base. If the question is simple and cannot be decomposed, the system will show the user a solution directly. Then the user can continue a new search. If the question this time is caused by several factors or can be represented in several ways, the system will show the user the most possible factor or the most possible representation. If the user confirms the factor or the representation given, the system will repeat the above procedure. Otherwise, the system will show the second most possible factor or the second most possible representation. Through this kind of interaction with the user, the system can find the solution in a convenient and accurate way.

When the system has matched the question that a user has input, automatically it will logically create an M-ary tree with the question as the root node. The

process of finding the solution is in fact a path from the root node to the leaf node.

The most possible factor or representation is dynamic and changing with the visits of the users. The simplest way is to control according to the number of users' visits. The most possible factor is one that is visited for the most times by users. The system will record the users' visits and will update knowledge-base in time. As for the same question, the next user may have a different visit path from the previous one. This is the self-learning process of the users' visit path.

If there is no question that can match a user's question, the system will automatically record the question in order to enlarge the knowledge-base.

To decrease the time complexity and space complexity of this system, when we build the knowledge-base, we choose m as 4, and the depth of the tree also as 4.

3.2. The transformation of M-ary tree to binary tree

In order to be more convenient and more rapid in storage and implementation, in the actual storage and visit, we will transform an M-ary tree into a binary tree.

The data structure of binary tree is as follows:

```
typedef struct BiTree{
   Relavant-Type data;
   Struct BiTree * leftpoint;
Struct BiTree * rightpoint;
}BiTree;
```

In the structure, the leftpoint points to the most possible factor or the most possible representation of the question. The rightpoint points to the next most possible factor or the next most possible representation.

If the question is a simple one, the leftpoint points to the solution to the question, and the rightpoint is NULL. The binary tree model is as Fig. 2 shown. MPF and MPRW are as the same meaning as in Fig. 1.

3.3. Algorithm description

Step 1: Input the question

Step 2: Through semantic interpretation, match the most similar question from knowledge base (the algorithm of sentence similarity computing is described in section 4). If the question input and the similar question do not exist in the knowledge-base, the system will record the question. And the search will stop. Otherwise, turn to step 3

Step 3: If the solution to the question exist, the system will show it to the users. Then the search stops. Otherwise, turn to step 4

Step 4: The system shows the most possible factor to the users. If the users confirm the factor, then turn to step 3. Otherwise, turn to step 5.

Step 5: The system shows the next most possible factor. If the users confirm the factor, turn to step 3. Otherwise, repeat step 5 until there are no contributing factors. Then the search stops.

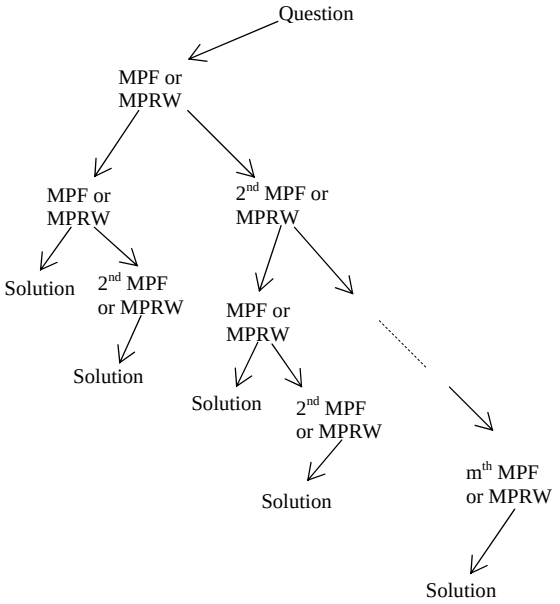

Fig. 2: Binary Tree Model

3.4. The main features

(1)Every path from the root node to a leaf node is a process of solving a question.

(2)Each path in a tree has a different weight which varies with the users' visits

(3)Tree creation begins with users' input of questions as root node.

(4)The theoretical model is M-ary tree, instead of a binary tree.

4. Algorithm for sentence similarity computing

Sentence similarity computing plays an important role in the Chinese QA systems based on knowledge-base. The matching of a question from knowledge-base is to find the most similar question to the target question from this set. The main method is to compute the similarity between each question in the knowledge-base and the target question. The question with the most similarity is the one that is being found. In this

paper, the following strategies are applied in sentence similarity computing:

(1) Computing based on words

(2) Increasing the weight of nouns and verbs in the sentence

(3) Computing comprehensively

(4) Using a synonym dictionary to increase semantic computing

4.1. Relative definitions and computing

Definition 1: Word Similarity WordSim(Q_1 , Q_2)

Sentence similarity is annotated through the form of sentence and word, thus indicating the similarity between sentences.

WordSim(Q_1 , Q_2) = 2×SameWord(Q_1 , Q_2) / ((Word(Q_1)+Word(Q_2))。

In this formula, SameWord(Q_1 , Q_2) indicates the number of identical key words in Q_1 and Q_2 .

If a key word appears for several times, it will be counted once. Question Words and words in the

Stop-word list are excluded, such as 为什么、怎么样、怎么、哪些、如何、的、地、得, ect.

Word(Q_i) indicates the number of key words in Q_i , i = 1,2

Notice: In our practice, we have found that nouns and verbs are very important in a sentence and nouns play a more important role than verbs. Sentences are built around nouns and verbs, so we have increased the weight of nouns and verbs in computing and have put the focus of sentences on the nouns and verbs. Thus, in counting the number of identical key words, if two words are identical and are both nouns, the number is counted as 5. If the two words are identical and are both verbs, the number is counted as 3. In the counting of the key words in Q_i , the nouns are also counted as 5, and verbs as 3. That is to say, a noun is counted as 5 if it appears once, and a verb is counted as 3 if it appears once. In programming g, after a sentence is segmented, Part-of-speech tagging of the words will be carried out in order to differentiate nouns and verbs.

Another point to pay attention to: Because a synonym dictionary is used, in the computing of sentence similarity, the weight of WordSim(Q_1 , Q_2) should be decreased. For example, the following two sentences are identical:

Q_1: 怎么杀计算机病毒？

Q_2: 怎么杀电脑病毒？

Definition 2: Sentence Length Similarity LenSim(Q_1 , Q_2)

Sentence similarity is annotated in terms of sentence length, thus indicating the similarity of sentence form to some degree.

LenSim(Q_1 , Q_2) = 1 – absolute-value(Len(Q_1) – Len(Q_2)) / (Len(Q1) + Len(Q_2))

In this formula, Len(Q_i) indicates the number of key words in Q_i, i = 1,2

Definition 3: Word Order Similarity OrdSim(Q_1 , Q_2)

Sentence similarity is annotated in terms of the order of key words, indicating the similarity of the position of the identical or similar words in the two sentences. This is weighed in terms of the number of synonyms or identical words that have reversed orders in the two sentences.

OrdSim(Q_1 , Q_2) = 1 – Rev(Q_1 , Q_2) / MaxRev(Q_1 , Q_2)

In this formula, MaxRev(Q_1 , Q_2) indicates the maximal reverse number of the natural number sequence of the number of the key words in $Q_1$and Q_2. For example, if there are 4 identical key words in Q_1 and Q_2 , the natural number sequence, is {4,3,2,1}, with a reverse number 6.

Rev(Q_1 , Q_2) indicates the reverse number of the natural number sequence that is made up of the position of the key words of Q_1 in Q_2.

The similarity of the position of synonyms or identical words in the two sentences is weighed in terms of the number of synonyms or identical words that have reversed orders in the two sentences. Suppose Q_1 and Q_2 are two questions, OnceWord (Q_1 , Q_2) is the set of synonyms or identical words in Q1 and Q_2 and repeated words will be counted as 1. P_{first}(Q_1 , Q_2) is the vector that is created by the order of the words of OnceWord(Q_1 , Q_2) that have appeared in Q_1(The vector is natural number order sequence and the position of a word is the position when it appears for the first time.). P_{second}(Q_1 , Q_2) is a vector that is created by the order of the words of P_{first}(Q_1 , Q_2) in Q2. Rev(Q_1 , Q_2) is the reverse number of the sequence of P_{second}(Q_1 , Q_2).

For example:

Q_1：我认为应该告诉你这个问题 Segmented：我 认为 应该 告诉 你 这个 问题

Q_2：告 诉 你 这 个 问 题 我 认 为 是 应 该 的 Segmented：告诉 你 这个 问题 我 认为 是 应该 的

The identical key words in Q_1 and Q_2 are OnceWord(Q_1 , Q_2)={我，认为，应该，告诉，你，这个，问题}.

In Q_1, the order of the key words is 我，认为，应该，告诉，你，这个，问题. The natural number sequence of this is{1，2，3，4，5，6，7}. The order of the seven key words in Q_2 is {5，6，7，1，2，3，4}. So RevOrd(Q_1 , Q_2) is the reverse number of {5，6，7，1，2，3，4} and its value is 12.

Notice: The identical key words in Q_2 and Q_1 are focused. The position of a word is the position when it appears for the first time.

Definition 4: Distance Similarity DisSim(Q_1 , Q_2)

Sentence similarity is annotated in terms of the distance of identical key words.

DisSim(Q_1 , Q_2)= 1 – absolute-value(SameDis(Q_1) - SameDis(Q_2)) / (Dis(Q_1) + Dis(Q_2))

In the formula, SameDis(Q_i) indicate the distance of the identical key words of Q_1 and Q_2 in Q_i , i = 1,2. If the key words are repeated, the maximal distance is counted.

Dis(Q_i) indicates the distance between the leftmost key word and the rightmost key word that are not repeated, i = 1,2. If the key words are repeated, the minimal distance is counted.

Definition 5: Sentence Similarity SenSim(Q_1 , Q_2)

It indicates the similarity between two sentences. The value is usually 0~1. 0 indicates no similarity and 1 indicates complete similarity. The larger the value is, the more similar the two sentences are.

Suppose two questions Q_1 and Q_2 and the similarity between them is SenSim(Q_1 , Q_2), then

SenSim(Q_1 , Q_2) = λ_1WordSim(Q_1 , Q2) + λ_2LenSim(Q_1 , Q_2) + λ_3OrdSim(Q_1 , Q_2) + λ_4DisSim(Q_1 , Q_2)

In the formula, $\lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 = 1$ and $\lambda_1 \geqslant 0.5 \geqslant \lambda_2 \geqslant \lambda_3 \geqslant \lambda_4 > 0$

The computing process is usually $\lambda_1 = 0.6$, $\lambda_2 = 0.2$, $\lambda_3 = 0.1$, $\lambda_4 = 0.1$. The similarity threshold is 0.65 when matching questions.

4.2. Algorithm description

Algorithm: A Method for Sentence Similarity Computing

Input: Two questions Q_1 and Q_2.

Output: the similarity of Q_1 and Q_2.

Step 1 Segmenting the two input questions Q_1 and Q_2 and obtaining two strings Q_1' and Q_2';

Step 2 Getting identical key words from Q_1' and Q_2';

Step 3 computing word similarity, sentence length similarity, word order similarity and distance similarity;

Step 4 computing the similarity between Q_1 and Q_2.

Compared with the computing methods of question similarity in other references, the key word retrieval in this method concerns sentence segmentation and pose of speech(other methods mostly concern sentence segmentation only). A synonym dictionary is needed when computing sentence similarity.

This computing method has the following features:

(1)It is simple and the surface information of a sentence is used.

(2)It has maintained the advantages of other computing methods and can guarantee the similarity when a clause or a phrase has moved in a sentence.

(3)It is more accurate than the others. The key words can partly express the syntactic information.

(4)It has added some semantic information by means of a synonym dictionary.

5. The building and maintenance of knowledge Base

The building and maintenance of knowledge-base are a prerequisite for the implementation of the QA system based on knowledge-base. The more information there is in the knowledge-base, the more effective the system is.

5.1. The source of knowledge

(1)The knowledge in the knowledge-base mainly comes from the following sources:

(2)Records of relevant departments

(3)Experiences of experts in the fields

(4)Knowledge gathered from the Internet

5.2. The updating of knowledge base

Knowledge-base should be updated on time. New information should be added to it timely. The main sources are:

(1)The newly-added records of relevant departments

(2)The new experiences of the experts in the fields

(3)The new knowledge that is created through manual processing then there is no matched question.

(4)New knowledge gathered from the Internet

5.3. The maintenance of knowledge base

The openness of the knowledge-base, having a unified standard interface and format.

Easiness to maintain Knowledge experts can maintain knowledge-base by some tools.

Modular. In order to fit the needs of different fields, the knowledge-base is designed as modules, and experts can only pay attention to the knowledge system in their fields.

6. The features of IICQAS

(1)Having made a breakthrough from the old method of simple answers

(2)Intelligent

(3)Interactive

(4)User-friendly

7. Experiments results

We have made our test in the knowledge-base that has more that 1000 pieces of knowledge. The parameter values are $\lambda_1 = 0.6$, $\lambda_2 = 0.2$, $\lambda_3 = 0.1$, $\lambda_4 = 0.1$. In the process of matching, the question that has the largest similarity (threshold >=0.65) is chosen.

After several interactions, the solution will be given in the end. If the maximal similarity is less than 0.65, it is considered that there is no solution to the question in the set. The accuracy rate in the test is more than 84%.

Three people select 100 questions to test respectively. The result of experiments is shown as Table 1.

Tester	Times of Test	Average Interaction Times	Accuracy Rate(%)
1	100	1.86	81
2	100	2.12	89
3	100	2.24	85

Table 1: Experiments Result.

8. Conclusions

In this paper, we have designed and implemented an interactive intelligent Chinese QA system based on a tree model and have provided an algorithm of interactive visits. This is a beneficial attempt in the development model of QA systems. On the other hand, the accuracy rate of sentence similarity computing is increased to a certain extent by means of key word retrieval, the use of the synonym dictionary and the increase of the weight of nouns and verbs in a sentence. However, we have not carried out a detailed analysis of the grammatical and semantic relations in a sentence. If all these are considered, the accuracy rate can be increased to a larger degree. This is the focus of our future research.

The interactive intelligent QA system has great prospects of application for customer services system, QA systems, etc. therefore, we will carry out our survey and research in this field.

Acknowledgement

This work is partially supported by National Nature Science Foundation of China (Grant No. 60675030).

References

[1] S. Li, B. Qin, T. Liu, et al, Overview of Question-Answering. *Journal of Chinese Information Processing*, 6:46-52, 2002. (In Chinese)

[2] C.B. Claude and R. Bertram, Research on Intelligent Question Answering System. *Technical Report*, May 1967.

[3] Z.H. Luo, X.Z. Fan, L. Liu. Ontology Used in the Automatic Question and Answer System. *Computer Engineering and Applications*, 32:229-232, 2005.(In Chinese)

[4] Y.J. Liu and Y. Xu, Automatic question answering system based on weighted semantic similarity model. *Journal of SourthEast University(Natural Science Edition)*, 5:609-612, 2004.(In Chinese)

[5] S.N. Qu, Q. Wang, Y. Zou, et al, Intelligent Question Answering System Based on Data Mining. *Journal of Zhengzhou University(Natural Science Edition)*, 2:50-54, 2007.

[6] Y. Zhao and Y.J. Liu, Research and Application of Semantic Relation in Intellegent Question Answer System, *Microcomputer Development*, 11:35-36(40), 2003.(In Chinese)

[7] J. He, An approach to generate boolean query in question and answering retrieval system. *Journal of Shandong University(Natural Science)*, 3:13-17, 2006.(In Chinese)

[8] S.X. Wang, Question Answering System: Core Technology, Application. *Computer Engineering and Applications*, 18: 1-3, 2005.(In Chinese)

[9] K.Y. Dai, S.S. Zhang and M. Wang. Research on Chat-Bot in Distributed Virtual Environment. *Computer Engineering and Applications*, 7:13-16, 2002. (In Chinese)

[10] B. Qin, T. Liu, Y. Wang, et al, Question Answering System Based on Frequently Asked Questions. *Journal of Harbin Institute of Technology*, 10:1179-1182, 2003.(In Chinese)

[11] Y.Z. Wu, J. Zhao, X.Y. Duan, et al, Research on Question Answering & Evaluation: A Survey. *Journal of Chinese Information Processing*, 3:1-13, 2005.(In Chinese)