行政院國家科學委員會專題研究計畫　期中進度報告

問答系統技術研發(1/3)－自然語言問答系統之研究

計畫類別：個別型計畫
計畫編號：NSC91-2213-E-002-097-
執行期間：91 年 08 月 01 日至 92 年 07 月 31 日
執行單位：國立臺灣大學資訊工程學系暨研究所

計畫主持人：陳信希

報告類型：精簡報告

處理方式：本計畫可公開查詢

中　華　民　國 92 年 5 月 19 日

1. Introduction

Question Answering (QA) becomes a hot research topic in recent years due to the very large virtual database on the Internet. QA is defined to find the exact answer, which can meet the users' need more precisely, from a huge unstructured database. Traditional information retrieval systems cannot afford to resolve this problem. On the one hand, users have to find out the answers by themselves from the documents returned by IR systems. On the other hand, the answers may appear in any documents, even that the document is irrelevant to the question.

Two possible approaches, i.e., keyword matching and template extraction, can be considered. Keyword matching postulates that the answering text contains most of the keywords. In other words, it carries enough information relevant to the question. Using templates is some sort of information extraction. The contents of documents are represented as templates. To answer a question, a QA system has to select an appropriate template, then fill the template and finally offer the answer. The major difficulties in this approach are to find general domain templates, and to decide which template can be applied to answer the question. Some other techniques are also useful. For example, to answer the questions "Who..." and "When ...", the identification of named entities like person names and time/date expressions will help to locate the answer.

In this report, we proposed three models, which integrate the information of Named Entity, inflections, synonyms, and co-reference. We plan to evaluate how each factor affects the performance of a question answering system.

2. Basic Question Answering System

The system is composed of three major steps: (1) preprocessing the question sentences, (2) retrieving the documents containing answers, and (3) retrieving the sentences containing answers.

2.1 Preprocessing the Question Sentences

Our main strategy is keyword matching. This approach has a drawback, i.e., the words used in the question sentences and in the sentences containing the answers may be different. For example, verbs can be in different tenses and synonyms can also be used. Therefore, we have to make necessary changes and expansions in the question sentences.

At first the parts-of-speech are assigned to the words in question sentences. Then, stop-words are removed. The remaining words are transformed into the canonical forms and selected as the keywords of the question sentences. For each keyword, we find all of its synonyms from WordNet 1.6. Those terms form an expansion set for the keyword. If the keyword is a noun, a verb, an adjective, or an adverb, all the possible morphological forms of the words in the expansion set are also added into this set. Here the morphological forms are the plural of a noun, different tenses of a verb, and the comparison of an adjective or an adverb. They are shown as follows:

noun AAA: AAAs | AA[s,z,sh]es

verb BBB: BBBed BBBing | BB[e]d BB[e]ing / BBBs | BB[s,z,sh]es

adjective or adverb CCC: CCCer CCCest | CC(y)ier CC(y)iest

The irregular nouns and verbs can be transformed by looking up the WordNet.

2.2 Retrieving the Documents Containing Answers

We implement a full text retrieval system to find the documents that may contain the answers. The purpose is to decrease the number of documents we have to search the answering sentences. Each keyword of a question sentence is assigned a weight, so are their various morphological forms. Those words tagged as NNP and NNPS, which denote proper nouns, have assigned higher weights. This is because they should be presented in the answer. The weights of added synonyms are less than the keywords. The score of a document is computed as follows:

$$score(D) = \sum_{t \in EX(T),\, t \text{ in } D} weight(T)$$

where T is one of the keywords, and $EX(T)$ its expansion set.

The document containing one keyword or any words in its expansion set earns a score of its weight.

Those documents that have scores no less than the threshold are selected as the answering documents. Threshold is set to the sum of weights of the words in the original question sentence. Note that the removed words have no scores. If no documents have scores greater than the threshold, we assume that no answers can be found for the question.

2.3 Retrieving the Sentences Containing Answers

Finally, we examine each sentence in the documents that may contain the answers. Those sentences that contain most words in the expanded question sentence are retrieved. The top five sentences are regarded as the answers. If there are more than five possible answers, we randomly select five of them. To meet the limit of 250 bytes, we truncate the sentences that exceed the limit. On the contrary, if the answer is shorter than the limit, we concatenate it with the next sentences.

3. Model Description

Three models proposed to see if expansion is helpful or not. Model 1 is a base model. Only inflections are added. Model 2 adds synonyms from WordNet (Miller, 1990). And Model 3 tries to resolve co-reference in a simple way. Each of them will be described in detail in later sections.

Besides, we select answers according to the named entities that the question might be relevant. Our QA system will guess the interested entity type by looking at the questions. Position of the interested answer terms is also important. If the length of answering sentences is longer than restricted length, the final answer text has to include the actual answer. We also propose a method to implement this idea. The proposed algorithm will be described later.

3.1 Interested Entity Type

After taking a question as input, our system first guesses which entity type the question is interested in. The method is simply rule-based. If the question starts with "who", "when", and "where", it may ask for a person name, a time/date expression, and a location name, respectively. If it starts with "what" or "which", or it is the "Name a ..."-type question, then the system goes on to look at the first noun behind it. We collected some keywords to indicate the interested entity types, such as "country" for location name, "person" for personal name, and so on.

3.2 Named Entity Extraction

Named entity extraction plays an important role in our experiments. It is introduced while deciding question focusing, doing question expansion, and measuring similarity between document passage and question sentence.

For named entity extraction, we employ several named entities dictionaries, such as gazetteer, a collection of family name, *etc*. Different from simply dictionary look-up, these dictionaries also include other useful information. For a personal name, we can know that it is a family name, a male first name, or a female first name. For a country name, we can get its adjective form as well as how to call its people.

For other location names, it provides the names of provinces or countries it belongs to as well. Organization names are accompanied by their abbreviations. We have not employed the information of types of personal names and the superior administrative division yet.

Time/date expression is simply keywords (Sunday, January, etc.) The resolution of expressions like “yesterday”, “last week” is still undergoing. Other named entities like quantity and numbers are not handled yet.

3.3 Base Model - Question Expansion by Named Entity and Inflection Forms

In Base Model, we first decide if there is a named entity in the question sentence. If so, we record its equivalence (e.g. abbreviation of an organization name). Notice that a named entity can be more than one word. For the rest words in the question sentence, we remove stop words and attach the root form and all the inflection forms of each of them. These newly invited terms are for the use of similarity comparison later.

The next step is to segment documents into passages as comparison units. The document set we use is the set of the 50 most relevant documents to the questions. The relevant document set is offered by NIST. In the Base Model, a passage is simply a sentence.

For each passage, we also identify named entities in it, but their equivalences are not attached. The inflections are not added either. This is because we have already introduced them in the question side.

Then we measure its similarity to the expanded question sentence. For each word (or phrase) occurs in the passage and also in the expanded question, it contributes a score to the similarity. By the recent experiment, if it is a named entity, it contributes 2 points; otherwise 1 point. If it occurs in the original question, the contributed score is doubled.

Besides, if a word (or a phrase) does not occur in the question but is of the interested type of the question, the FOCUS tag is set and the position of this word is recorded.

While giving answers, those words (or phrases) that are assigned the FOCUS tag are reported first. The passage of higher score is considered to be more possible to carry the answer and is ranked higher.

To meet the length restriction, we have to truncate the passages longer than 250 bytes. We decide the focusing center of each answering passage first. Truncate characters 125 bytes ahead of the center and also the exceed part if the remaining passage is still longer than 250 bytes. For those assigned a FOCUS tag, the center is the average position of all the found named entities of interest. For those did not, the center is the average position of words that also occur in the question sentence.

3.4 Model 2 – More Expansion by Synonyms

Besides the basic structure of Base Model, we also expand questions by the synonyms of ordinary nouns or verbs, i.e., those which are not named entities. Synonyms are obtained by looking up the WordNet (Miller, 1990).

3.5 Model 3 – Passage with Co-Reference Resolved

This model is also based on the Base Model. But we want to resolve co-reference problem first before measuring similarity with the question sentence. We proposed a simple strategy to do so: take the first sentence as a passage. If the next sentence contains pronouns (except "it"), it is merged into the previous passage. Or if the next one contains a phrase of the pattern "the A" and the word "A" occurs in the previous passage, it is merged into the previous one, too. It can help resolve anaphora problem as well as the co-referential noun phrases.

4. Evaluation

Table 1 lists the results of our three models. Three runs were evaluated. Each run is for each model, i.e., qantu01 for Base Model, and so on. Each answer text can be judged as Wrong, Correct, and Unsupported. "Unsupported" means that the document associated to the answer text does not really support the answer. The Strict Evaluation only counts Correct ones, and the Lenient Evaluation takes both Correct and Unsupported ones as correctly answered.

Table 1. Results of Three Models in the QA Systems

Run ID	Strict		Lenient		Strict (Debugged)	
	MMR	Failed	MMR	Failed	MMR	Failed
qantu01	0.315	377 (55.3%)	0.348	354 (51.9%)	0.333	368 (55.0%)
qantu02	0.315	376 (55.1%)	0.341	354 (51.9%)	0.327	365 (53.5%)
qantu03	0.278	394 (57.8%)	0.309	370 (54.3%)	0.284	394 (57.8%)

By Table 1, half of the questions failed to be answered. It is better than the case that we only answered 1/3 of the questions correctly. There are 24 more questions in average answered by unsupported documents. Comparing the performance of different models, Base Model and Model 2 are almost the same, but Model 3 is worse than the other two. Model 2 answered one more question than Base Model did, but Base Model offered unsupported answers at higher ranks than Model 2 did in the Lenient Evaluation. Model 3 is worse in either evaluation.

It seems that adding synonyms does not help a lot. It even lows down the speed. The most difficulties we met in QA are often paraphrases, not only synonyms. Therefore, it might be more efficient to tackle the paraphrases problem.

The reason that Model 3 worked badly may be the over-simplified co-occurrence resolution. For those questions failed to be answered here but successful in the other two runs, it was often the case that the passages containing the answer texts have been expanded into large ones. The occurrence of co-reference candidates is too frequent to simply concatenate sentences.

But co-reference resolution is helpful for question answering. During the investigation, we found that a portion of questions can be answered by keyword matching with co-reference resolved. To integrate the co-reference resolution part into the system, or find an alternative way to tackle it will be another important future work.

5. Conclusion

This paper proposed three models. These models can help us to see the usefulness of each proposed factor. Base Model uses the information of named entity and its equivalence, as well as the information of inflection forms of general nouns and verbs. Synonyms of nouns and verbs are proved to be of little use. Simple co-reference resolution causes a drawback because of the wrongly merged passages.

References

Lin, C.J. and Chen, H.H. (1999) "Description of Preliminary Results to TREC-8 QA Task." *TREC-8 Proceedings*, Gaithersburg, Maryland, http://trec.nist.gov/pubs/trec8/papers/ NTU_TREC8_QA.pdf

Miller, G. (1990) "Five Papers on WordNet." *Special Issue of International Journal of Lexicography* 3.

MUC (1998) *Proceedings of 7th Message Understanding Conference*, http://www.muc.saic.com /proceedings/proceedings_index.html.

Salton, G. and Buckley, C. (1988) “Term Weighting Approaches in Automatic Text Retrieval.” *Information Processing and Management*, Vol. 5, No. 24, 513-523.