Tianwang at TREC-2006 QA Track

Jing He, Yuan Liu
Network and Distribution System Laboratory
School of Electronic Engineering and Computer Science
Peking University
{hj,liuyuan}@net.pku.edu.cn

Abstract

This paper describes the architecture and implementation of Tianwang QA system2006, which works for the TREC QA Main task this year. The main improvement is: 1. add one well founded knowledge source from Web – Wikipedia, and employ some natural language processing technologies to extract high quality answers; 2. design and implement a new translation algorithm in query generation. The results show that fine organized knowledge source is effective in answering all three types of questions. And such query generation algorithm can be benefit from both Frequent Asked Questions on Web and past TREC QA data.

1. Introduction

As the second time participating in TREC QA task, we reuse the system developed last year[1]. As many other researchers pointed out, knowledge source performs well in Question Answering task, we experiment the effectiveness of this approach. However, different from Web knowledge, which is large in quantity and high in duplication, knowledge from a source maybe high in quality but limited in quantity. So we have design a different strategy to extract answers. Another improvement is in query generation. Last year, we designed a iterative approach to generate Boolean queries. However, both the number of iteration and the loosing/contraction strategy is not established. We build a translation model to generate query this year, collaborating with the approach last year.

This paper is organized as below: In section 2, the overview of Tianwang QA system2006 is described. In section 3, the detail processing technology is described. The query generation and use of Wikipedia are described in section 4 and section 5, separately. In section 6, we give the conclusion and future work in this field.

2. System Overview

The system is similar to that of last year. However, because we are lack of high quality Knowledge Base source and Frequent Asked Question data source, we replace them with another type of knowledge source – Wikipedia. So, there are five modules in our system: data processing, question analysis, answer extractor and answer integration/validation, as Figure 1 shows.

We can see clearly from the figure that the system includes two parts: offline part and online part. The offline part preprocesses the collection dataset and index them, generating inverted files for searching online. Unlike the system last year, we do not put a lot of work such as crawling KB and FAQ data and indexing them in offline part. The reason for this is two-folded: first of all, the knowledge base of Wikipedia contains large quantity of pages, most of which are not useful in this

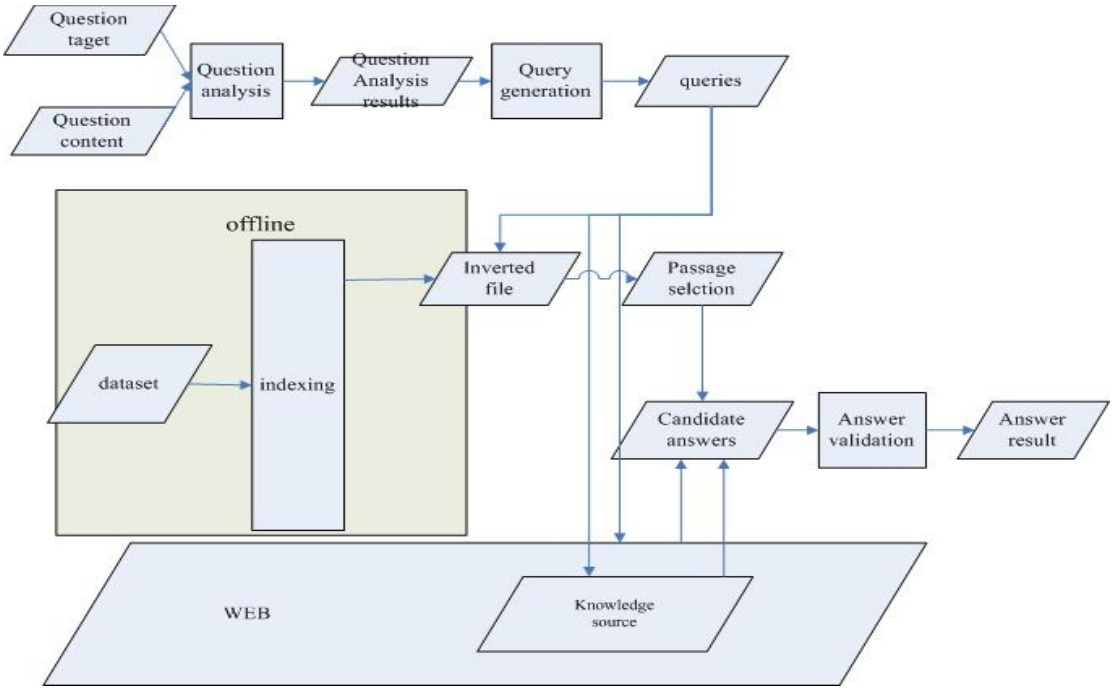

Figure1: Tianwang QA System 2006 infrastructure

TREC task, and crawling them is time-consuming; another reason is the data source (web site) has supplied good interface for accessing the information online. Most of modules are similar to those of last year, except query generation and knowledge source modules. The detail of processing is described below, in section 3.

3. Question Answering Processing

In question answering processing, question classification is the first step. The architecture of categories follows a two-layered question taxonomy, containing 6 coarse grained and 50 fine grained categories. Each coarse grained category contains a non-overlapping set of fine grained categories. As we did last year, we use the SVM method with one against all strategy and bag of words feature to perform the classification[10]. Each question is labeled with only one category with maximum probability. The training database is provided by UIUC, which contains 5,500 labeled questions[9]. Because the training data does not fit to QA-test-set this year, precision is not so good as last year and then constraint the performance of later modules.

Then the candidate relative passages are retrieved from both collection and Web resources. Web sources are important to QA task, so we choose search engine and Wikipedia as our source candidates. Three search engine are selected, including Google, Yahoo! and MSN. For each question, we select key words from it and forward them to the search engine, and then store the result for future use. Wikipedia is a more important source, the content of it is thought to be more confident than other web pages. We use the "target" as the basic element to retrieve page, and then make fine analyses to it, described in section 5.

At last, we should extract answer to each question. High weight is assigned to the result extracted from Wikipedia. If the result from Wikipedia is existed , the result will be returned immediately. If not, then we try to extract answer from TREC document passages and search engine snippets. We label entities in these contents. The entity type include GATE entities and some predefined entity type. We assign each entity a score and rank the entities according to the scores.

The score computation is similar to [9]. It mainly based on its frequency of occurrences in the documents and snippets. Then we check whether the type of the highly ranked entity's type matches the question's category. In most cases, the matching is boolean. But there are some exceptions. Some types of question such as TIME/YEAR/DATE can match with a ranked list of answer types.

However, the answer extracted from other information sources, may not be located in the AQUAINT data source, leading to unsupported result, so we should filter them to avoid such errors. If a entity is filtered, the next candidate entity is validated. The processing continues until a candidate answer is validated to be appear in suitable passage or no more candidate answer in the set. In the second situation, a NIL will be given. For list questions, the first 5 entities are taken as answers default; for factoid questions, the first entity is as the answer and for definition questions, we implement a simple summary algorithm to summarize the main ideas of top 5 passages retrieved by passage retrieval sub-module.

4. Translation Model for Query Generation

Translation model is a famous module in the field of machine translation[2][3], employed to represent the relation of query and text content in Information Retrieval[4] and Question Answering[5] these years. Because we can get the question and answer data in past TREC QA tasks. What is more, we also can make use of FAQ data we crawled last year. All these data are training data for building a translation model.

First of all, we extract most frequently words in questions, removing some stop words and question type words such as what, which, when, etc. Then we need to calculate the transferring probability from one of these terms to another, which is useful to generate the queries. The probability is described as:

$$P(t \mid s) = \lambda^{-1} \sum_{i=1}^{N} c(t \mid s, J^i)$$

$$c(t \mid s, J^i) = \frac{P(t \mid s)}{P(t \mid s_1) + P(t \mid s_2) + \quad + P(t \mid s_n)} \#(t, J^i) \#(s, J^i)$$

Where t and s are two different terms, $P(t \mid s)$ is the probability of transforming from term s to t, λ is a factor for normalization. J^i is a pair of question and answer. $\{s_1, s_2, \ldots, s_n\}$ is the term set in J^i, $\#(t, J^i)$ and $\#(s, J^i)$ are count of t and s in J^i. Obviously, it is a process of iteration. The possibility of transforming can be thought to be a possibility of appearance of term t in relative passage when term s appears in question. Therefore, it's rational to take a term with high transforming possibility as query term.

5. Wikipedia as Knowledge Source

The Knowledge base and FAQ perform not so well last year because the data is too sparse, so they can only answer questions in limited domain. Wikipedia, which is a popular knowledge source in Web[6][7], supplies much more knowledge than the knowledge source we employed last year. We can see it can cover more than half targets in QA task each year. Therefore, it is a good choice to use it.

Though we may be sure the answer exists in Wikipedia articles, it's difficult to extract them . Because the duplication is not as high as in web, the approach of simply counting named entities we employed last year will not effective. Instead, we use some NLP technologies to help finding right answers. For this, we count the frequency of question types in past TREC QA tasks, select the frequency asked question type, and construct answer patterns for them, according to [8]. So we may match the exact answer in Wikipedia by these patterns.

6. Conclusion and Future Work

The main improvement of Tianwang QA system 2006 is to refine the process of query generation and extend the knowledge source. The results show the accuracy (MAE) is similar to last year, though the question is declared to become more difficult (time dependent questions). However, the rate unsupported answers increases this year. Some other particulars also found this problem. The main reason of such increase may be that though answers can be found in knowledge source effective, we are lack of a good approach to validate it in document itself.

There are much more work to do in future. First of all, the approaches should be measured carefully in quantity way in next step. What is more, we would like to focus on some new task such as CiQA next year. It is also very interesting for us to answer more than factoid questions, such as questions about subjective feeling and features of a data set.

7. Acknowledgments

This work described therein is supported in part by PRC Ministry of Education grant 20030001076 and by NSFC grant 60435020.

Reference

[1] J.He et al.(2005)."Tianwang at TREC-2006 QA Track," In *Proceeding of TRE*C 2005.

[2] P.Brown, J.Cocke, S. Della Pietra, V.Della Pietra, F. Jelinek, J.Lafferty, R.Mercer, and P.Roossin(1990)."A statistical approach to machine translation," *Computaional Linguistics*, 16(2), pp.79-85.

[3] P.Brown, S.Della Pietra, V.Della Pietra, and R.Mercer(1993)."The mathematics of statistical mechine translation: Parameter estimation," *Computational Linguistics*, 19(2), pp.263-311.

[4] A.Berger and J.Lafferty(1999)."Information retrieval as statistical translation," In *Proceeding of the 1999 ACM SIGIR Conference on Research and Development in Information Retrieval*,pp.222-229

[5] J. Jiwoon, W. B. Croft, and L. Joon Ho, "Finding similar questions in large question and answer archives," in *Proceedings of the 14th ACM international conference on Information and knowledge management*. Bremen, Germany: ACM Press, 2005.

[6] E Hovy, U Hermjakob, CY Lin(2001),"External Knowledge Sources for Question Answering," In *Proceeding of TREC* 2001.

[7] Jinxi Xu, Ana Licuanan and Ralph Weischedel(2003), "TREC 2003 QA at BBN: Answering Definitional Questions,"In *Proceeding of TREC* 2003.

[8] Deepak Ravichandran and Eduard Hovy(2002). "Learning surface text patterns for a Question Answering system," In *Proceedings of the 40th ACL conference*. Philadelphia, PA.

[9] X. Li and D. Roth. "Learning Question Classifiers," In *Proceedings of the 19th International Conference on Computational Linguistics (COLING'02)*, 2002.

[10] D. Zhang and WS Lee (2003) "Question Classification using support vector machines," In *Proceeding of SIGIR* 2003