Complex Question Answering with ASQA at NTCIR 7 ACLIA

Yi-Hsun Lee[1], Cheng-Wei Lee[12], Cheng-Lung Sung[1], Mon-Tin Tzou[1], Chih-Chien Wang[1], Shih-Hung Liu[1], Cheng-Wei Shih[1], Pei-Yin Yang[1], Wen-Lian Hsu[1§]

[1]Institute of Information Science, Academia Sinica, Taiwan, R.O.C
[2]Department of Computer Science, National Tsing-Hua University, Taiwan, R.O.C
§corresponding author

{rog,aska,clsung,mttzou,vincent,journey,dapi,annabel,hsu}@iis.sinica.edu.tw

Abstract

At NTCIR 7, we implemented the Academia Sinica Question Answering (ASQA) system for complex questions. The system uses three methods to select answer strings from a news corpus. (a) It uses syntactic patterns, which are usually used by QA systems, to retrieve more precise answer strings than those derived by traditional IR. (b) Using external knowledge, the system can find accurate answers to specific questions that the traditional IR approach can not process. (c) Entropy-based and co-occurrence-based mining methods are used to retrieve relevant answer strings for document retrieval. In the NTCIR 7 CCLQA task, ASQA achieved 0.26 in the CT-CT task and 0.20 in the CS-CS task.

1. Introduction

Because of the high level of information overload on the Internet, research into question answering, which focuses on how to respond to users' queries with exact answers, is becoming increasingly important. In recent years, many international question answering contest-type evaluation tasks have been held at conferences and workshops, such as TREC [3], CLEF [1], and NTCIR [2].

At NTCIR 7, the systems were required to process complex questions related to definitions, biographies, relationships, and events. This year, our system focused on external knowledge, syntax patterns and some data mining techniques. We introduce the methods in Section 5.

The remainder of this paper is organized as follows. Section 2 provides an overview of the system architecture. In Sections 3 and 4, we introduce the question analysis strategy and document retrieval method, respectively. In Section 5, we discuss the methods used at NTCIR 7. In Section 6, we consider the redundancy removal module. Section 7 evaluates the system performance, and Section 8 contains a discussion. Section 9 summarizes our conclusions.

2. System Description

Our question answering system ASQA (Academia Sinica Question Answering) for complex questions is divided into five modules: question processing, document retrieval, sentence selection, answer ranking, and redundancy removal. Questions are first analyzed by the question processing module to extract the keywords and question topic, which are then used to retrieve documents from the corpus. In the next step, the sentence selection module obtains the sentence scores, after which the sentence ranking module determines whether or not the sentences are relevant. Finally, the redundancy removal module filters similar sentences as answer strings.

3. Question Analysis

The question processing module uses simple surface patterns to classify questions into four types: (a) definition, (b) biography, (c) relationship, and (d) event. For example, biography questions always start with "誰是," which we call a question term.

The main topics of such questions are usually adjacent to the question terms, so we use surface text patterns to retrieve a question's topic. In biography

questions, we consider that the question topic and the question keywords should be the same.

In the other question types, we use a keyword extractor to extract keywords from question topics. To ensure that a question is not divided into meaningless small pieces by the keyword extractor, we retrieve Wikipedia titles to address the problem. If the question topic appears in the Wikipedia titles, we set the keywords as the question topic.

4. Document Retrieval

The document retrieval module uses Lucene, an open source information retrieval engine, to index documents by character. In the retrieval process, we use the "AND" operator to form a query string comprised of question topics, which is then sent to the Lucene engine. If the number of retrieved documents is larger than a threshold, we terminate the retrieval process; otherwise, we use the keywords to form a new query string. If the number of retrieved documents is still less than the threshold, we use the "OR" operator to combine the keywords as a query string. After retrieving the documents, we split each one into several sentences.

5. Sentence Selection

The sentence selection module uses co-occurrence-based and entropy-based methods, syntactic patterns, and external knowledge to find relevant sentences. We describe the process in detail in following sub-sections.

5.1. Co-occurrence-based Sentence Selection

Magnini et al. [5] consider that the number of documents in which the question terms and the answer co-occur is useful for QA. The hypothesis is similar to that of Clarke et al. [4], who use co-occurrence methods to measure the relevance of an answer to the given question based on Web search results. Although the method is used in factoid QA, we think the hypothesis could also be useful for complex QA.

Our rationale is that, in good quality passages, the more often a term co-occurs with the question topic, the higher the confidence that the passage will be relevant. We regard a co-occurrence as an indication that the passage can be taken as correct based on the co-occurring question topics.

Let the given term be t and the given question be Q, where Q consists of a set (QT) of question terms {qt1, qt2, qt3, ……, qtn} created from the question processing result of Q. We use the following equation to calculate each term's weight:

$$p(t_i \mid Q) = frequency(t_i \cap Q) / freuqency(Q)$$
$$p(t_i) = frequency(t_i) / N \quad ,$$

where P (ti|QT) denotes the probability of the term i when QT appears; N is the number of documents in the corpus; and P(ti) is the probability of the term i occurring in the news corpus. We consider that the probability P (ti|QT) should be significantly different to the probability P(ti). After calculating the probabilities of the terms, we use the following hypothesis test to determine whether or not the term is relevant:

$$H_0 : p(t_i \mid Q) = p(t_i)$$
$$H_1 : p(t_i \mid Q) > p(t_i) \quad .$$

5.2. Entropy-based Sentence Retrieval

Inspired by the work of Xu and Croft [6], we use the concept of entropy, which is a measure of uncertainty in information theory, to perform sentence retrieval in our question answer system. This concept is similar to the inverse document frequency (IDF) in information retrieval. If the entropy value of a term is large, the term is regarded as a stop word or keyword in some topic. For example, the term "political party" should be important in political news.

The entropy value of each term we use is calculated by:

$$Entropy(t) = -\sum_{i=1}^{n} p(t_i) \times \log_{10} p(t_i) \qquad (1)$$

Let t and n denote the given term and the number of retrieved documents respectively. Then, the probability of t is calculated by

$$P(t_i) = frequency(t_i) \Big/ Total(t) \qquad (2)$$

where $frequency(t_i)$ represents the number of passages i in which the term t occurs, and $Total(t)$ denotes the number of retrieved passages in which t occurs.

Next, we select relevant sentences based on their entropy values. We adopt the vector space model to calculate the similarity between the retrieved passages and an entropy vector, which is constructed from the terms whose entropy values are greater than a threshold. In our system, the threshold is set at 0.6. In addition, we use the IDF to filter stop words. The sentence vector is constructed based on the segmentation result. Each weight of the vector represents the term's entropy value or a default value. Because the entropy method is data-driven, we only use it if the question topic appears in several documents.

5.3. External-Knowledge-based Sentence Retrieval

There are several knowledge databases, such as WordNet, HowNet, and Wikipedia. The knowledge they contain should be useful for QA systems. To support our ASQA, we use Wikipedia, an online encyclopedia whose content is provided by Internet users. If the keywords of a question appear in a title in Wikipedia, the contents of the title should be the standard answers that also appear in the corpus. In addition, we consider the case where the question does not appear on a Wikipedia page. To address this problem, we employ the three methods described in the following sub-sections.

5.3.1. Related Term-based Sentence Retrieval

In this sentence retrieval method, we also use entropy values to find related terms. A low entropy value indicates a topic-specific term, which should be useful for identifying answer sentences. However, in our experience, terms with small entropy values are not useful when the corpus is derived from news articles or the Web, because Web data or corpus data contains too much noise. Therefore, we only apply the entropy method to the Wikipedia data because it contains less noisy information. For example, in Wikipedia "Lesley" who is the daughter of "Ma Ying-jeou" only co-occurs with "Ma Ying-jeou," but on the Web, "Lesley" co-occurs with many terms that are not of interest to us. Obviously, this sentence selection method can only be used when the question topic exists in Wikipedia.

Equation (1) is used to calculate each term's entropy value. Then, we apply the following equation to calculate the sentence score:

$$Score(s)\begin{cases}\beta & \text{If } entropy(term_i)=0 \\ \sum_{i=1}^{m} 1/entropy(term_i) & \end{cases} \quad (3)$$

The smaller a term's entropy value, the more relevant the term will be. If a sentence contains many relevant terms, it will be taken as the standard answer. Therefore, we aggregate the inverse of the terms' entropy values as the score of s.

5.3.2. Definitional Term-based Sentence Retrieval

Some terms or phrases provide useful hints for identifying definitional sentences. For example, the phrase "is a kind of" is a clear indication that the sentence is about the type or category of an object or term. Since such phrases or terms are used a great deal in Wikipedia, we collect the Wikipedia pages to extract such phrases or terms, which we call "definitional terms".

This sentence selection method uses Equation (1) to find definitional terms. In contrast to the method described in sub-section 5.3.1, which utilizes small entropy values to locate related terms, we want to find terms or phrases with large entropy values that occur in most Wikipedia articles. However, these kinds of terms could also be stop words or definitional terms, so they must be filtered out. We assume that stop words are evenly distributed in any corpus. If a term has a high entropy value or a high IDF value in both the Wikipedia corpus and the News corpus, the term is probably a stop word. Finally, we calculate the score of a sentence by aggregating the scores of terms that have high entropy values as follows:

$$Score(s)\begin{cases}0 & \text{If } entropy(term_i)<threshold \\ \sum_{i=1}^{m} entropy(term_i) & \end{cases} \quad (4)$$

5.4. Syntactic Surface Pattern-based Sentence Retrieval

Syntactic patterns are useful for finding precise information for biography questions. Next, we describe how we construct biographical syntactic surface patterns (SPs).

Table 1. TREC Biography Question Type

Q Types	Year 2003	Year 2004	Year 2005	Year 2006	Sum	percentage
Appellation	24	7	0	2	33	5.76%
Ancestral Home	1	1	0	0	2	0.35%
Nationality	2	0	0	0	2	0.35%
Domicile	1	0	0	2	3	0.52%
Birth/Death Day and Place	8	4	11	14	37	6.46%
Race	1	0	0	0	1	0.17%
Family	7	2	7	10	26	4.54%
Parentage	3	0	0	0	3	0.52%
Occupation	56	7	6	22	91	15.88%
Education	0	0	3	2	5	0.87%
Belief	21	3	1	2	27	4.71%
Characteristics	10	0	5	14	29	5.06%
Honour	18	15	10	17	60	10.47%
Contribution	45	15	30	26	116	20.24%
Notables	10	10	0	5	25	4.36%
Organization	9	5	0	9	23	4.01%
Quotations	0	1	1	0	2	0.35%
Publications	6	1	12	25	44	7.68%
Disease	2	5	0	2	9	1.57%
Disability	2	0	1	0	3	0.52%
Misadventure	4	2	2	12	20	3.49%
Scandal	12	5	4	4	25	4.36%
Controversy	5	1	3	5	14	2.44%
Others	16	37	53	11	117	20.42%
					717	

5.4.1. Syntactic Pattern Generation

To construct SPs for biography questions, we need To construct SPs for biography questions, we need categories of biographical information. Since 2008 was the first year that TREC considered biography questions, we had to use English training data for our analysis. We took the test questions from TREC 2003 to TREC 2006 and categorized them into 24 types. Some of the question types were combined based on their similarity, which resulted in the 20 types shown in Table 1. Then, from the biographical information in the table, we created our own Chinese training questions and answer nuggets to train the SPs.

We created the SPs semi-automatically from the following sources.

1. Alignment-based Template Generation: For relational information, such as the relationship between a husband and wife, we collected the relational term pairs and applied our alignment-based template generation method to obtain preliminary sentence patterns. Then, high frequency patterns were rewritten into answer templates by our analysts.
2. Sentence making: We tried to think of possible biographical answers for each type ourselves and then wrote corresponding templates.
3. Internet: Definitional sentences in biographical articles obtained from the Internet, such as Wikipedia, were also referenced during template formation.
4. News Corpus: Our analysts used the Lucene search engine to extract articles about the targeted person from the CIRB20 and CIRB40 corpora, and then selected relevant definitional sentences in the articles

Table 2. Biography Template Type

Biography20Attributes	Instance	Rule	TOTAL Rules
Appellation|稱謂稱號	0	15	15
Belief|理念信仰	0	8	8
Characteristics|個人特色	0	7	7
Contribution|貢獻影響	0	9	9
Controversy|爭議	0	0	0
Domicile|居住地	1	4	5
DPBD&Nationality|生卒年地與國籍	11	29	40
Education|教育背景	0	17	17
Family|家族成員	0	29	29
Honour|獎章榮譽	3	16	19
IPC|病症與肢障	1	8	9
Misadventure|不幸事件	0	5	5
Notables|重要人物	0	9	9
Org&Occ|組織與專業	0	38	38
Others|其他	0	6	6
Parentage|家世背景	0	1	1
Publications|相關著作	0	6	6
Quotations|名言語錄	0	4	4
Race|種族民族	0	4	4
Scandal|醜聞	0	4	4
TOTAL	16	219	235

to make corresponding templates.

5.4.2. Negation Term

To help screen out unwanted sentences while keeping the good ones, we added two categories of keywords: "Terms to be excluded" and "Terms that should not be excluded." Most of the content of the first category is comprised of negative terms, while the second contains double negatives. The following are some examples of the two types of terms; the words in parentheses are the literal meanings of the Chinese terms:

1. Terms to be excluded: 拒絕(reject)、禁止(ban)、不能(can't)、不會(won't)、應該(should)、別想(Don't think about)、嗎(questioning particle)、謠傳(rumor)、毋須(not necessary)、…。

2. Terms that should not to be excluded: 不得不(can't help but…)、不能不(can't…without…)、無不(no..not...)、無非(no other than...)、莫不(none...not...)…。

Punctuation

In Chinese, the sentence structures of declaratives and questions are often the same. If a question does not have a questioning particle, such as "嗎" or "呢", it would be very difficult for a computer program to distinguish it from a declarative. Therefore, a sentence that matches our templates may be just a question, rather than a declarative that we wish to retrieve. To avoid this problem, all sentences ending with question marks were deleted when retrieving sentences, unless the question marks were part of the matched templates.

6. Redundancy Removal

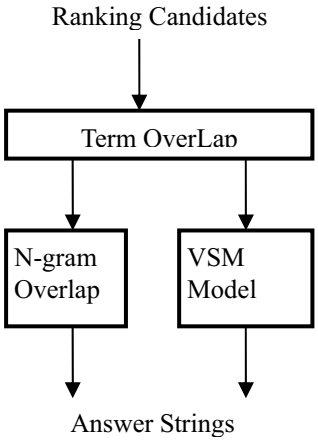

Figure 1 Redundancy Removal FlowChart

A news corpus may contain a great deal of redundant information, especially if the news is hot, because similar information could be mentioned in different news articles. To handle this problem, we need a module to filter out similar sentences.

We adopt a two-stage redundancy removal approach, as shown in Figure 2. First, we use the concept of term overlap to remove similar sentences. The sentence selection module identifies relevant terms, definitional terms, and entropy terms. In this stage, we set a new threshold to find good terms from the set of terms derived by the sentence selection module. It is assumed that these terms would be the same in similar sentences.

In the second stage, if we have answer candidates that match some of the syntactic patterns described in sub-section 5.4, we use an N-gram overlap method to remove similar sentences; otherwise, we use a vector space model to remove such sentences.

7. System Performance

In the NTCIR 7 CCLQA Task, we achieved an F-score of 0.2666 in CT-CT and 0.2034 in CS-CS. Among the four question types, our system achieved a higher F-score for biography questions than for the other three types. The system did not perform well on event questions because we only used IR techniques to retrieve relevant sentences.

All of our system's modules are based on traditional Chinese; therefore we have to convert simplified Chinese questions and corpora. However, as shown by the results, the process degrades the system's performance.

	ALL	Best
IASL-CT-CT-01-T	0.2666	0.2666
IASL-CS-CS-01-T	0.2034	0.4X

8. Discussion and Error Analysis

Our analysis of the experiment results identified certain problems that need to be addressed. We discuss them in the following sub-sections.

8.1. Segmentation of NER problems

Because we only used a character-based index for document retrieval and did not use an NER technique to filter out wrong documents with different people's names, we encountered some problems when information about the wrong person was retrieved. For example, there was a question "誰是李寧？". However, there was another person called "李寧遠" in the news corpus. Examples like this cause the performance to deteriorate on some questions.

8.2. Different Perspectives between Training and Test Data

According to the training data provided for the task, most of the answers were summarization-oriented. The answers contained a lot of related information about the question focus, and they were quite different from the answers for factoid questions. Unfortunately, we found a discrepancy between the training data and test data because some of the test questions were judged as if they were "factoid" questions. For example, there was a question "誰是李安？". In the training set, the answers to this kind of question would contain information like birthplace, birthday, family or occupation, but there is only a small portion of this information in the standard answers in the test data. The above-mentioned discrepancy had a substantial impact on the system's performance.

8.3. Inconsistency in Treating People with Same Name

For questions like "誰是李寧？" and "誰是舒馬克？", there is more than one person with the same name in the news corpus. We found that the task evaluators treated them in different ways. For "李寧", there are at least five people with that name in the news corpus, but only the person who does gymnastics was labeled as correct. However, for "舒馬克", all the answers related to different "舒馬克" were labeled as correct. Some errors

were caused by this inconsistency.

8.4. Unmarked Correct Answers

System responses judged as correct answers should be marked, yet some correct answers were unmarked. This degraded our system's performance substantially. For example, the standard answer "日本唱將型歌手", which appeared in our system response "小柳由紀屬於唱將型歌手," was unmarked.

9. Conclusion

For NTCIR 7 CCLQA, we built a complex question answering system. Since the task is "complex," we assume the answers are summarization-oriented, which means they contain various types of information that requires careful filtering. We used several syntax patterns to select precise answers when dealing with biography questions, and used the entropy method to select definitional terms and related terms from Wikipedia for definition questions and relation questions respectively. Even though the pattern-based methods were not highly accurate, we still encountered the low coverage problem. Therefore, we adopted the co-occurrence and entropy-based methods to find relevant sentences.

10. Acknowledgments

This research was supported in part by the National Science Council of Taiwan under Center of Excellence Grant NSC 95-2752-E-001-001-PAE; and by the Research Center for Humanities and Social Sciences, Academia Sinica, and the Thematic Program of Academia Sinica under Grant AS 95ASIA02.

We wish to thank the Chinese Knowledge and Information Processing Group (CKIP) of Academia Sinica for providing us with AutoTag for Chinese word segmentation.

11. References

[1] Cross Language Evaluation Forum (CLEF), http://www.clef-campaign.org/

[2] NTCIR Workshop, http://research.nii.ac.jp/ntcir/

[3] Text REtrieval Conference (TREC), http://trec.nist.gov/

[4] C. L. A. Clarke, G. Cormack, G. Kemkes, M. Laszlo, T. Lynam, E. Terra and P. Tilker, Statistical Selection of Exact Answers (MultiText Experiments for TREC 2002), *Proc. of TREC*, 2002, pp. 823–831.

[5] B. Magnini, M. N. R. Prevete and H. Tanev, Is it the right answer?: exploiting web redundancy for Answer Validation, *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, 2001, pp. 425-432.

[6] J. Xu and W. B. Croft, Query expansion using local and global document analysis, *Proceedings of the 19th annual international ACM SIGIR conference*, Zurich, Switzerland, 1996, pp. 4-11.