NTT DATA TREC-9 Question Answering Track Report

Toru Takaki
Research and Development Headquarters
NTT Data Corporation
Kayabacho Tower Bldg., 1-21-2, Shinkawa,
Chuo-ku, Tokyo 104-0033 Japan
E-mail: takaki@rd.nttdata.co.jp

Abstract

This paper describes the processing details and TREC-9 question answering results for our QA system. We use a general information retrieval strategy and a simple information extraction method with our QA system. Two types of indices, one for documents and one for passages, were used for our experiment. We submitted four results, two for each category of short and long answers. A score of 0.231 for the short category and 0.391 for the long category was obtained.

1 Introduction

Question-Answering (QA) processing has been attracting a great deal of attention recently. This type of retrieval processing requires the use of techniques to retrieve pertinent information from within a document that differ from those used for document retrieval. A QA system that can retrieve concise and suitable information that satisfies the needs of users will contribute to the improvement of recall precision and enhance a user's productivity when they are searching for information using the vast and ever expanding resources of the worldwide web. The currently used document retrieval method, which outputs a document list, forces users to search individual documents to find the information they desire. The Text REtrieval Conference (TREC) is designed the QA Track as one of tracks from TREC-8 in 1999. We regard QA as an application that unites natural language processing, information extraction, and information retrieval processing, which is a traditional and refined technology that is based mainly on the frequency of term occurrence. This traditional information retrieval technology and natural language processing, based on semantics, are indispensable to the QA processing.

We participated in the TREC-9 Question-Answering track held this year. This was the second time we have participated in a QA track (TREC-8 was the first). For this track, we combined the traditional information retrieval technique and the information extraction technique to construct a QA system. Our TREC-9 QA system was based on our TREC-8 QA system to which we had made some improvements. Our official runs at the TREC-9 QA were executed by changing some parameters and the units of the index, document, or passage to be retrieved in the initial retrieval processing that are applied in traditional information retrieval. In this report, our QA system is described along with an analysis of the initial search-processing step, and the results of our official runs are shown.

2 Processing flow

This section describes the processing flow of our QA system. The processing was done according to the four steps below (see Figure 1):

(1) Question analysis
The answer type is specified by an analysis of question. Then, query terms for the initial search are extracted from the question sentence,

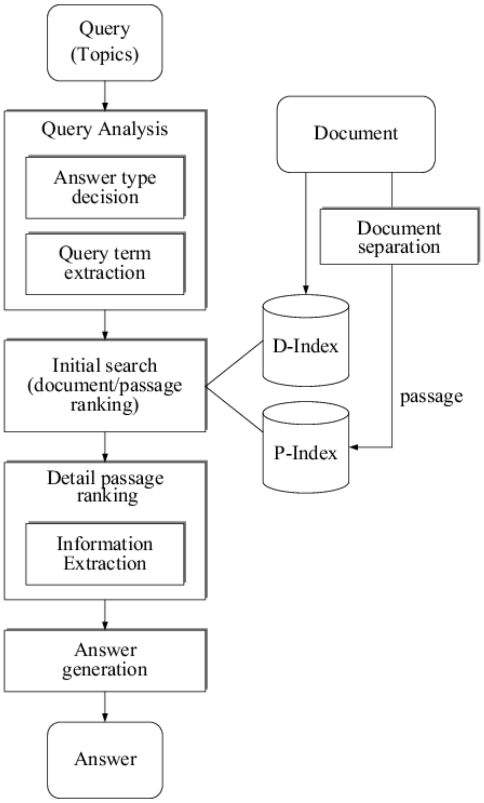

Figure 1: Processing flow

(2) Initial search
An initial search is done to limit the number of documents searched in the next step of the processing. The traditional ranking method, based on term frequency, is applied to the initial search,

(3) Detail passage ranking
Selection of important passage-spans and their rankings are done. The passage-span ranking uses the information extraction results of the specified answer type, and

(4) Answer generation
To provide an answer within the restricted length, 50 bytes or 250 bytes, the answer is extracted from the top-ranked passage.

Processing details are described below.

2.1 Answer type specification

The answer type specification is a processing step in which a determination is made as to what type of answer is required for a given question (topic). This specified answer type is used to extract the answer from the document in the next step. In the answer type specific processing, we created question templates that defined the answer type for question phrases, such as a *wh-determiner*, a *wh-pronoun*, etc. For instance, Topic No.206 "***How far away is the moon?***" suits the template "***How far ....***" , so the answer type is determined to be LENGTH. We defined 28 answer types, as shown in Table 1. These answer types have a hierarchical structure. Therefore, one question is will not always have only one answer type; sometimes two or more answer types can be given. In the case of Topic No.271 "***How tall is a giraffe?***", the prime answer type candidate is LENGTH and the second candidate is NUMBER, that is in a high-ranking hierarchy. The determination of whether or not the answer type of a high-ranking hierarchy is to be an answer type candidate is made based on a consideration of the question template. Moreover, questions that have no template match are given UNDEFINED as their answer type. Our template does not have provisions for a why-question.

2.2 Query term extraction

The query terms are extracted from the question, and used to search the candidate documents in document database. The purpose of this search is to minimize the number of the candidate documents. The high-cost processing executed later, such as passage ranking and information extraction, is done for only documents where the probability is high that they include the correct answer. The search for ranking is based on the frequency of the query term, like "ad-hoc" retrieval, and the top-ranked document is the candidate having the correct answer. However, the query term expansion processing usually done in an "ad-hoc" retrieval is not performed in our system.

Top level	Middle level	Bottom level
PROPER	PERSON	CHAIRMAN
		LEADER
		MINISTER
		PRESIDENT
		SECRETARY
		SPECIALIST
	LOCATION	CITY
		COUNTRY
		STATE
	COMPANY	
	LAKE	
	RIVER	
	MOUNTAIN	
	LANGUAGE	
NUMBER	SIZE	
	LENGTH	
	MONEY	
	PERCENT	
	PERIOD	
TIME	DATE	
	YEAR	
UNDEFINED-PROPER		
UNDEFINED		

Table 1: Answer type

Deletion of stop words

Unnecessary terms were deleted from a question in accord with a 550-stopword list.

Extraction of multiword phrase

A multiword phrase was extracted by using a part-of-speech tagger and then used as the query term. Each single-word term, which was parts of the multiword phrase, was also made into a query term.

Extraction of preposition phrase

In the QA retrieval, some questions required limited information. Topic No.32 used in TREC-8 QA "*What is the largest city in Germany?* " required the "*largest city in Germany*". If "*in Germany*" is not extracted as a query term, other "*largest city*", such as "*in the world*" or "*in Japan*" etc., cannot be distinguished without "*in Germany*". Therefore, the preposition phrase is important in the QA retrieval. Thus, the preposition phrase was made a query term.

Extraction of quotation phrase

Since a quotation phrase is a limiting expression that is close to the content of a question, like a preposition phrase, we adopted it as a query term.

Query term's weighting

The degree of importance was given to a basic word, a multiword phrase, a quotation phrase and a preposition phrase. The multiword phrase is divided into single-words, and both the single-words and the multiword used as query terms.

Unit of index of retrieval document

The document set used for TREC-9 consists of the following data sets from the TIPSTER and TREC document CDs:
AP Newswire (Disks 1-3),
Wall Street Journal (Disks 1-2),
San Jose Mercury News (Disk 3),
Financial Times (Disk 4),
Los Angeles Times (Disk 5), and
Foreign Broadcast Information Service (Disk 5).

In our experiment, the following two units were used as an index for the initial search.

(1) Original document (data was the part enclosed with <DOC >and </DOC >)
978,952 documents of TREC-9 evaluation used as index units.

(2) Paragraph divided the original document
The unit of division was different depending on the kind of the document. More than 978,952 documents were divided into 11,343,632 parts.

The QA track required the extraction of the pertinent answer, not the unit of document. So the division of document into a paragraph by paragraph ranking made it possible to extract a more suitable answer. Each paragraph is given an identifier, such as AP90424-0079-000046, that are a combination of the paragraph identifier (000046) and the document identifier (AP900424-0079).

Ranking of initial search

The query terms and their weights are input into the initial search. Both the document and the paragraph are ranked according to the input. In our QA system, we did the relevance ranking of a document or a paragraph using the BM25 function of Okapi. This function is as follows:

$$\sum_{T \in Q} w^{(1)} \frac{(k_1+1)tf}{K+tf} \frac{(k_3+1)qtf}{k_3+qtf} \qquad (1)$$

where
Q is a query, containing terms T,
$w^{(1)}$ is the Robertson/Sparch Jones weight of T in Q,

$$w^{(1)} = \frac{(r+0.5)/(R-r+0.5)}{(n-r+0.5)/(N-n-R+r+0.5)} \qquad (2)$$

N is the number of documents/paragraphs in the collection,
n is the number of documents/paragraphs containing the term,
R is the number of documents/paragraphs known to be relevant to a specific question,
r is the number of relevant documents/paragraphs containing the term,
K is $k_1((1-b)+b \times dl/avdl))$,
tf is the frequency of occurrence of the term within a specific question,
qtf is the frequency of the term within the question from Q was derived, and
dl and $avdl$ are the document length and average document length.

The documents or paragraphs that ranked in this ranking process are considered to include the correct answer. Therefore, for the subsequent processing, we used only the top-ranked documents from the document ranking and the documents that included the top-ranked paragraph.
Both the top m_d documents from the document ranking and the top m_p documents of the paragraph ranking were assumed to be the following processing object. Even if there is an overlapping of the top m_d and m_p documents, m_d and m_p are not changed. The number of following processing documents to be processed subsequently is assumed to be M.

2.3 Passage ranking and information extraction

The candidate passages that may include the answer are specified and extracted from the M top-ranked documents obtained in the previous step. These passages are part of the top-ranked documents, and the part has much the query words/terms and the words/terms matched the answer types. By using this concept there is a high probability of finding the correct answer. This passage extraction method is based on traditional information retrieval techniques, such as relevance ranking. A passage was extracted using the following procedure:

(1) Scoring by query term
A score was given to each word of the top-ranked document. This score was based on the inversed document frequency (IDF) measure of query term q_i. When each word of document D was assumed, the word P_i (i = 1,2,3,...) from the top of the document in ascending order, a score, $IDF(q_i)$, given to word P_j where query term Q had appeared. Moreover, score IDF' is given to the word of P_k at the circumference term position of P_j, and this score was based on the distance from P_j. The longer the distance from P_j, the smaller the score given to the circumference word. In some cases, where there were two or more query terms in the question sentence, a term score was given to each query term, and the sum of the score given by each query term was assumed to be the score for P_j. The consecutive passages where the score was more than the threshold were determined to be candidate passages.

(2) Scoring by answer type
Same as the scoring by query term. The bonus score for each word in the extracted passage was given by the words of the answer type. The answer

type was given one or more candidate types in the order of importance.

The word of these answer types was extracted by the information extraction technique, and the bonus score was given to the word. When the maximum value of the bonus score of the word of the prime candidate's answer type is S, S/k was given to the word of the k-th candidate's answer type. In our information extraction, we prepared proper name dictionaries, such as country, city, world region, U.S. state, currency, a personal name, and the dictionaries of literal form, such as date and time.

2.4 Answer generation

An answer generator outputs the answer string within the restricted length number, from 50 up to 250 bytes. The region, which included the word having the highest score in the passage, was outputted as the answer. The system did not output the string same as the term within the question.

3 Analysis of initial search

In this section, we analyze the initial search, which is one of QA retrieval steps used by the topics of the TREC-9 QA track as evaluation data. The initial search is based on a traditional information retrieval technique. Here, we analyze the change of the initial search accuracy by using the index for document or paragraph units.

Initial search

The QA initial search is a relevance ranking of the document/paragraph used with the query terms that are extracted by using the question sentence. In the QA retrieval, some natural language processing and information extraction processing are necessary, and the cost of this processing is needed. Placing restrictions on the amount of data to be searched is useful from the viewpoint of the processing speed, especially when retrieving a huge amount of data. Moreover, the use of traditional information retrieval technology is also beneficial. However, there is a method of whereby the information extraction result can be put in the index beforehand. In this method, if the information extraction module is imperfect, the information extraction processing for all data to be indexed must be done after the module is corrected. Therefore, we adopted this method of initial search.

Initial search accuracy by difference of index

We analyzed the initial search ranking that show how many document had the correct answer in the document top-ranked by the TREC-9 QA question and dataset. In this analysis, we used the TREC-9 QA judgment file provided by NIST, the top 1000 ranked document results ranked by the AT&T version of SMART provided by NIST, and our initial search results that were used for our submitted systems. We investigated the highest ranked document that included the correct answer, outputted by each system's initial search for each TREC-9 QA question. High precision is required in a QA retrieval, especially so in the rules of the TREC QA (it is not required that a system output all the correct answer phrases in a document). In addition, this analysis becomes the indicator of the threshold decision for how many top-ranked documents should be used to obtain the highest accuracy. The initial search retrieval results of our system and the SMART system were examined and the highest ranking, which contained the correct answer for each question, were examined. Table 2 shows the number of the question at the highest rank that included the correct answer for 682 TREC-9 QA topics. Here, the percentage shows the accumulation ratio of a ranking.

We prepared an index of both the unit of the document and for each paragraph so as to perform a comparison. NTTD-D means by document index and NTTD-P means paragraph index. In NTTD-D, the document of 48.2%, 67.7%, 75.7%, and 80.5% contained the correct answer of a question at the rankings of 1,3,5, and 10. Even for rank 5, the rising degree of the accumulation ratio was high but the rising growth was lower at the lower ranking. The tendency of SMART was also similar. Moreover, the retrieval accuracy of document index (NTTD-D) was better than that of the para-

Highest rank	SMART		NTTD-D		NTTD-P	
	#Q		#Q		#Q	
1	287	(42.1%)	329	(48.2%)	279	(40.9%)
2	63	(51.3%)	83	(60.4%)	90	(54.1%)
3	39	(57.0%)	50	(67.7%)	45	(60.7%)
4	33	(61.9%)	39	(73.5%)	16	(63.0%)
5	33	(66.7%)	15	(75.7%)	29	(67.3%)
6	19	(69.5%)	6	(76.5%)	13	(69.2%)
7	10	(71.0%)	11	(78.2%)	10	(70.7%)
8	7	(72.0%)	6	(79.0%)	7	(71.7%)
9	5	(72.7%)	6	(79.9%)	15	(73.9%)
10	3	(73.2%)	4	(80.5%)	11	(75.5%)
11-20	39	(78.9%)	31	(85.0%)	42	(81.7%)
21-30	21	(82.0%)	18	(87.7%)	21	(84.8%)
31-40	18	(84.6%)	11	(89.3%)	11	(86.4%)
41-50	10	(86.1%)	12	(91.1%)	9	(87.7%)
51-60	7	(87.1%)	6	(91.9%)	3	(88.1%)
61-70	4	(87.7%)	6	(92.8%)	6	(89.0%)
71-80	4	(88.3%)	3	(93.3%)	1	(89.1%)
81-90	3	(88.7%)	3	(93.7%)	2	(89.4%)
91-100	2	(89.0%)	1	(93.8%)	4	(90.0%)
other	75	(100.0%)	42	(100.0%)	68	(100.0%)

Table 2: Highest rank of initial search

graph index (NTTD-P) in the comparison of the initial search. In our system, the parameter setting of the BM25 function for the document index did not change for paragraph index in the initial search. Therefore, it was thought that this was the reason for the decrease in accuracy for the paragraphs. However, we did not do a detailed analysis. Moreover, it would have been necessary to analyze whether to or not the paragraph division was done correctly. We set the threshold, that is the number of the document to be used for processing after initial search, to 5 or less in our TREC-9 QA system, and the used document is very limited.

4 TREC-9 evaluation result

We submitted four results in TREC-9 QA track; there are two results each for the 50-byte answer and 250-byte answer categories. **NTTD9QAa1S** and **NTTD9QAa2S** are run names for the 50-byte category, and **NTTD9QAa1L** and **NTTD9QAb1L** are for the 250-byte categories. The difference for each run are the index used and the number of the top-ranked document used for the detail passage ranking in the initial search. As mentioned above, the sum document of m_d, from the document index, and m_p, from the paragraph index, were used as candidate documents to do the detail passage ranking. The parameter was $m_d = 3$ and $m_p = 2$ in **NTTD9QAa1S** and **NTTD9QAa1L**, $m_d = 5$ and $m_p = 3$ in **NTTD9QAa2S**, and $m_d = 3$ and $m_p = 0$ in **NTTD9QAb1L** (using only the document index and not the paragraph index). The other processing was the same for each run. Table 3 summarizes the evaluation results provided by NIST for our system. The results show that our mean reciprocal rank (MRR), except **NTTD9QAa1S**, was better than the average of all participants. We calculated the difference of MRR with **NTTD9QAa2S** and the average for all participants in the 50-byte category, and analyzed our system with having large MRR difference, named

Run tag name	Mean reciprocal rank (MRR) [Average MRR]	Num. of answers found at rank X						#Q Best	#Q ≥ Med
		1st	2nd	3rd	4th	5th	Not found		
NTTD9QAa1S	0.216 [0.218]	103	49	34	23	12	461	158	597
NTTD9QAa2S	0.231 [0.218]	108	61	24	26	24	439	161	599
NTTD9QAa1L	0.391 [0.350]	191	95	51	35	11	299	208	541
NTTD9QAb1L	0.381 [0.350]	195	79	40	26	29	313	212	534

Table 3: Our submitted TREC-9 QA runs

	Topic No.	MRR of **NTTD9QAa2S**	MRR of Average	Difference	Question
1	817	1.000	0.030	0.970	*Boxing Day is celebrated on what date?*
2	633	1.000	0.061	0.939	*How long do hermit crabs live?*
3	490	1.000	0.061	0.939	*Where did guinea pigs originate?*
4	541	1.000	0.061	0.939	*What was the purpose of the Manhattan project?*
5	779	1.000	0.080	0.920	*Name the university of which Woodrow Wilson was president.*
6	731	1.000	0.091	0.909	*What amount of folic acid should an expectant mother take daily?*
7	383	1.000	0.098	0.902	*What is the largest variety of cactus?*
8	661	1.000	0.119	0.881	*How much does one ton of cement cost?*
9	398	1.000	0.121	0.879	*When is Boxing Day?*
10	815	1.000	0.121	0.879	*What is the date of Boxing Day?*

Table 4: Best results and questions

as best and worst, shown in Tables 4 and 5.

First, we analyzed the answer type decision procedure. When our best and worst were compared, there were a lot of "*where*" questions in the worst. In the "*where*" question (71 questions), **NTTD9QAa2S** was 0.235 against the average MRR of 0.314. The reason for this low performance, determined by a detailed analysis of the "*Where*" question results was that either the answer types CITY, COUNTRY, or STATE were judged as a LOCATION. LOCATION is a more abstract answer type. Therefore, another feature extraction that can judge in detail and another answer type should be added to our system. However, as for 59 questions judged to be answer type NUMBER, **NTTD9QAa2S** result was excellent; the MRR was 0.260 vs. 0.201 for the average MRR. Next, the initial search result was analyzed. It was apparent that all top-ranked documents of **NTTD9QAa2S**'s initial search included the correct answer in the best case. For the worst case, we examined the 10 worst questions and found only one question for which the initial search failed to give a document including the correct answer a high ranking to document. This shows that our system failed in either the passage ranking or answer generation steps. In the case of Topic No.614 "*Who wrote the book, "Huckleberry Finn"?*", the correct answers are "*Samuel Langhorne Clemens*" and "*Mark Twain*". The correct answer was included in a document of the second rank in an initial search of NTTD-D and of the first rank in NTTD-P. However, our system has a problem in that it is not able to extract the correct answer phrase in the 50-byte answer category when the answer appeared at a position away a little. This reason is that our system emphatically determined the important part of a passage using the appear-

	Topic No.	MRR of **NTTD9QAa2S**	MRR of Average	Difference	Question
1	474	0.000	0.717	-0.717	*Who first broke the sound barrier?*
2	859	0.000	0.606	-0.606	*Where is Rider College?*
3	614	0.000	0.602	-0.602	*Who wrote the book, "Huckleberry Finn"?*
4	270	0.000	0.601	-0.601	*Where is the Orinoco?*
5	363	0.000	0.589	-0.589	*What is the capital of Haiti?*
6	698	0.000	0.588	-0.588	*Where is Ocho Rios?*
7	495	0.000	0.579	-0.579	*When did Aldous Huxley write, "Brave New World"?*
8	727	0.000	0.578	-0.578	*Where is Procter & Gamble based in the U.S.?*
9	440	0.000	0.574	-0.574	*Where was Poe born?*
10	378	0.000	0.573	-0.573	*Who is the emperor of Japan?*

Table 5: Worst results and questions

ance density of the query term. Another problem of our system is that a correct answer cannot be consistently acquired; the phrase before and behind that is occasionally extracted. Thus, it was necessary to use linguistic information that cans more detailed extraction in the answer generation part in restricted length.

5 Summary

We described our TREC-9 QA processing system and discussed the result of our experimental retrieval searches. It was determined from an analysis of the data that the results of our initial search were roughly excellent result. However, we found that even if an initial search is successful, the correct answer could not always be correctly extracted. Our results suggested that the correct answer could be extracted roughly by a traditional information retrieval technique in the QA retrieval, but that natural language processing and the information extraction processing are indispensable for a complete extraction. We will examine the application of linguistic processing and information extraction to a QA retrieval technique using a key phrase within the question sentence in the future.

References

[1] S.E. Robertson, S. Walker, and M. Beaulieu, Okapi at TREC-7: automatic ad hoc, filtering, VLC and interactive, in *Proc. of the Seventh Text REtrieval Conference (TREC-7)*, NIST Special Publication 500-242, pp.253 - 264, 1999.

[2] T. Takaki, NTT DATA: Overview of system approach at TREC-8 ad hoc and question answering, in *Proc. of the Eighth Text REtrieval Conference (TREC-8)*, NIST Special Publication 500-246, pp.523 - 530, 2000.

[3] E. Voorhees, The TREC-8 Question Answering Track Report, in *Proc. of the Eighth Text REtrieval Conference (TREC-8)*, NIST Special Publication 500-246, pp.77-82, 2000.