POSTECH Question-Answering Experiments at NTCIR-4 QAC

Seung-Hoon Na, In-Su Kang , Jong-Hyeok Lee
Div. of Electrical and Computer Engineering
Pohang University of Science and Technology (POSTECH)
Advanced Information Technology Research Center (AITrc)
San 31, Hyoja-Dong, Pohang, South Korea, 790-784
{ nsh1979,dbaisk, jhlee}@postech.ac.kr

Abstract

This paper describes our system and additional experimental results in NTCIR-4 QAC Task 1. The main components of our system are question classification, passage retrieval, and named entity extraction. Passage retrieval was performed by a density-based ranking method based on importance of query terms occurred in the passage. Question classification and Named entity extraction were designed by the rule-based approach that uses lexico-semantic patterns, in which the Kadokawa thesaurus is basic semantic resource. Our QA system consisting of these basic components showed a weak performance in NTCIR-4, but obtained high performance in pure answer extraction.

Keywords: *Question Answering, Passage Retrieval, Named Entity Tagging, Answer Extraction*

1 Introduction

Approaches to the question answering system are divided into two categories: deep-level approaches [2,3,4,6,7,11] and shallow-based approaches [1,5,13,14]. First, deep-level approaches focus on performing deep language processing, such as parsing and semantic analysis, and generate correct answers by strong answer justification. To successfully construct deep-level approach, a large knowledge base should be constructed because linguistic representations acquired by deep language processing is complex and very specific, causing the system not to match the representation of question and representation of context sentence in the correct answer [11]. Although top ranked QA systems in TREC is based on deep-level QA, it is a critical limitation that the construction of such a system is very expensive.

Shallow-based approaches are very popular, because its system is based on a relatively simple framework and enables rapid development and provides domain-adaptability, but also shows a comparative performance against deep-level approaches [13,14]. For these reason, our system adopts shallow-based approaches.

The three main components of a shallow-based system are question classification, passage retrieval and, named entity classification. Passage retrieval is the module that locates the candidate answer in top retrieved passages. It is critical since the failure of passage retrieval forces the QA system to generate incorrect answers. Question classification and named entity classification are also important modules that the type of named entity is used as key clue for answer extraction step to match its entity type and semantic type of question.

In our system, passage retrieval was implemented by ‘density-based scoring’, the state of the art ranking family reviewed in [15]. For question classification and named entity classification, lexico-semantic patterns are constructed that can achieve high matching coverage for each rule by using semantic information, but also present specific lexical-level rules based on lexical word. The representation of our lexico-semantic patterns is very similar to [13], but our NE classification is slightly different from [13] in the sense of that the classification process is cascaded into *simple tagging*, *relationship-based tagging*, and *partial matching* steps. Kadokawa thesaurus is a reference semantic hierachy for constructing lexico-semantic patterns. The semantic similarity between concept codes and concept-level matching is used in our QA system.

The remainder of this paper is organized as follows. In section 2, we describe our overall QA system architecture and its details. In section 3, we report the evaluation results of our system. Finally, section 4 concludes this paper.

2 QA System Architecture

Figure 1 describes our QA system. The system consists of four components – *Question analysis*, *document retrieval*, and *passage retrieval* and *answer extraction*. First, question analysis analyzes a question to determine its answer type and formulates a query for document retrieval and passage retrieval. Document retrieval and passage retrieval retrieves top relevant documents and top relevant passages respectively, in which correct answer could be contained. Finally, candidate answers are generated by answer extraction.

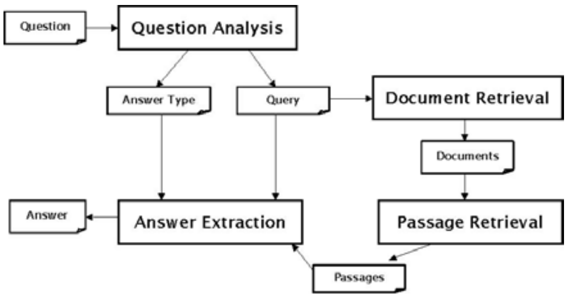

Figure 1. QA system architecture

ALL
- PERSON
- ORG
  - COMPANY
- LOC
  - COUNTRY
  - CITY
- BUILDING
- UNIT
- MONEY
- DATE
- ACRONYM
- FULLNAME
- TITLE
  - MOVIE
  - DRAMA
  - MUSIC
  - AWORD
  - MEANING
- SUBSTANCE
- ANIMAL
- PRODUCT
  - CAR
  - SOFTWARE
- ENAME

Figure 2. Answer type taxonomy

2.1 Question Analysis

In question analysis, a user's given question is classified into its semantic category in our answer type taxonomy. The answer type taxonomy consists of hierarchical structure, which is constructed by analyzing 200 questions NTCIR-3 QAC task, as described in Figure 2.

To classify a question into an answer type on the taxonomy, we constructed lexico-semantic patterns as in Table 1. Each lexico-semantic pattern takes the tagging results by Chasen [8] as an input , which is

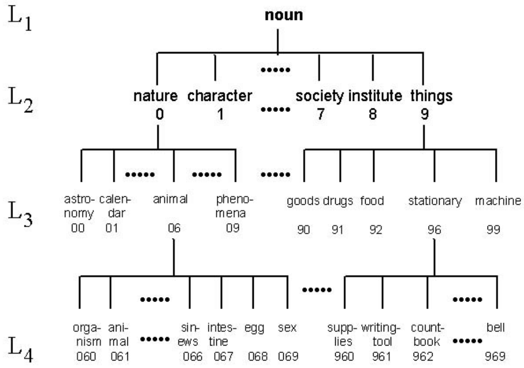

Figure 3. Concept hierarchy of the Kadokawa thesaurus

matched by regular expression described in the pattern. A lexico-semantic pattern consists of lexico-semantic units, which basically comprised of three-fields; POS tag, lexical word and concept code. '*' indicates the wildcard to match an arbitrary substring of the input, and surrounding special characters '[' and ']' are used to identify 'subtype' of the question. Here, subtype is more specific categorical information rather than answer type. Each concept field consists of a POS part and concept part. For example, the concept field 'n5%' consists of POS part 'n' and concept part '5%'. This concept code information covers all lexical words that belong to hypo-concepts of concept code 5 (level 2) in Kadokawa thesaurus [10], which has a 4-level hierarchy of about 1,100 semantic classes, as shown in Figure 3. Concept nodes in level L_1, L_2 and L_3 are further divided into 10 subclasses.

For example, consider the question ″日本人として７人目の大リーグ選手となったのは誰ですか。″. The tagged results by Chasen is as following.

日本人/ CMCN + として/ fjc + ７/ CMS + 人/ CMSD + の/ fjC + 大リーグ/ CMPORG + 選手/ CMCN + と/ fjc + なっ/ YBD + た/ fYB + の/CT + は/ fjk + 誰/CTP + です/ fYB + か/ fjb + 。/g

Pattern	Sample question	Answer type	Subtype
＊ _/誰 ＊	″８９年に自殺した青木伊平氏は誰の秘書でしたか。″ (" Idaira" who suicided at 1989 is whose secretary?)	PERSON	
＊［ CMCN|CMD ］_/は _/何 ＊	"日本初の火星探査機は何でしょう。″ (What is first Mars Explorer of Japan?)	ENAME	探査機
＊［ _/_/n5% ］＊ _/は _/誰 _/です ＊	″日本人として７人目の大リーグ選手となったのは誰ですか。″ (Who is seventh Japanese player in Major Leaque Baseball?)	PERSON	選手
＊ _/いつ _/です _/か ＊	″クローン羊のドリーが誕生したのはいつですか。″ (When was clone sheep Dolly born?)	DATE	
＊［ _/_/n84% ］_/に _/は _/どんな ＊	"ミレーの作品にはどんなものがありますか。″ (What is the name of pictures by Millet?)	TITLE	作品

Table 1. Examples of lexico-semantic patterns for classifying questions

<table>
<tr><th>Pattern</th><th>Sample sentence</th><th>Answer type</th><th>Subtype</th></tr>
<tr><td>< CMP/$isalpha* CMP|CMPORG > CM%/%社/n713%</td><td>マーチン 社</td><td>Company</td><td></td></tr>
<tr><td>CMS+ [CMSD]</td><td>８２センチ
２ 回</td><td>Unit</td><td>センチ
回</td></tr>
<tr><td>[CMCN] g/「|『 < C%|f%|Y%|gO%|gS%|gC%*
CM%|Y%+ > g/」|』</td><td>民間衛星「イコノス」</td><td>Title</td><td>衛星</td></tr>
<tr><td>g/ (< _/$isalphaspace+ > g/)</td><td>ＮＩＨ</td><td>Acronym</td><td></td></tr>
</table>

Table 2. Examples of lexico-semantic patterns for named entity tagging

Here, ‘CMCN’, ‘fjc’, and ‘CMS’ and ‘CMPORG’, etc. indicate acronyms for tag-set used in Chasen. After tagging, question classifier sequentially tries to match lexico-semantic patterns for the given question, and then the following lexico-semantic pattern (third row in Table 1) will be matched.

* [_/_/n5%] * _/は _/誰 _/です * => PERSON

In our Japanese dictionary, a lexical word ‘選手’ has the sense that corresponds to concept code 559 in the Kadokawa thesaurus. As a result, above question is successfully matched, and answer type is PERSON and ‘選手’ becomes the subtype of the question.

After question classification, query for retrieval is generated by extracting root-forms of nouns and verbs from tagged results.

2.2 Document Retrieval and Passage Retrieval

Document retrieval finds the most relevant documents by the query generated from question analysis. Our document retrieval is performed using Okapi’s BM 25 formula [12].

Passage retrieval is a critical issue in question answering, as mentioned by [15]. We identify all passages in each document in top 100 documents retrieved by document retrieval module. A passage is defined to be *K* consecutive sentences and must include at least one query term. Passages are scored as idf-summation of query terms occuring in each passage. Passage definition and ranking method equals to the modified version of SiteQ ranking method in [13].

2.3 Answer Extraction

Answer extraction is most important component in question answering [1], which consists of three processing steps; named entity tagging, recognition of candidate answer, ranking candidate answer.

2.3.1 Named Entity Tagging

In named entity tagging, all entities occurring in the top passages are recognized and classified into concepts in answer type taxonomy. Named entity tagging step consists of three-processing steps; simple tagging, relationship-based tagging, and partial matching.

First, for simple tagging, basic entities such as person, organization, and location can be acquired from the result of Chasen. However, Chasen does not find all basic named entities in given documents, so rules for finding basic entities were constructed. Rules for entities with other semantic types are also constructed together. Our rules and examples for named entity tagging are described in Table 2. These rules are almost the same as rules used in question analysis. Named entity boundary can be described using a special symbol ‘<’ and ‘>’. ‘*$XXX*’ is special predicate function that is applied to single tagged lexical unit.

After simple tagging, relationship-based tagging is performed. Patterns for relationship-based tagging can include tagged results acquired from single tagging.

Finally, we apply partial matching to detect co-reference relation among entities in a document. The motivation of partial matching is diversity of context information to classify the same entity. For example, consider a part in a document below.

In JA-991011062
［視聴率］すずらんとあすか
…
ＮＨＫ連続テレビ*小説*「すずらん」が終了しました 。

In the document ‘JA-991011062’, ‘すずらん’ occurred twice. At the first occurrence, there is no clue information to infer its semantic category in the local context. By contrast, at the second occurrence, there is clue information such as ‘小説’ and title marker, so the entity ‘すずらん’ is classified into type TITLE with subtype ‘小説’. By partial matching, the entity of the first occurrence is matched with the second occurrence, so the entity is also classified into the same type as the second occurrence. This partial

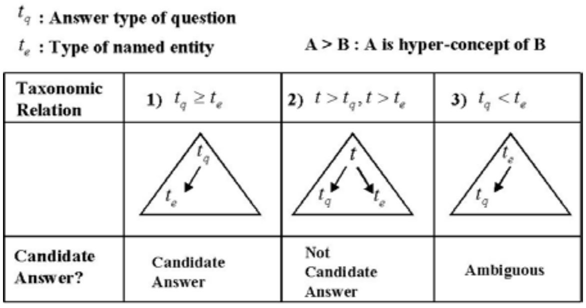

Figure 1. Three taxonomic relations for recognition of candidate answers

matching scheme is known to be useful, increasing recall of named entity tagging [9].

2.3.2 Recognition of Candidate Answer

Given a set of entity types and the answer type, we decide whether the given entity is a candidate answer or not, using taxonomic relations between the entity type and the answer type. There are three possible relations as in Figure 4.

If a taxonomic relation between the entity type and the answer type belongs to the first case $t_q \geq t_e$ or the third case $t_q < t_e$, then the entity is regarded as a candidate answer. However, when the taxonomic relation is the third case, the possibility of the candidate answer to be a correct answer decreases according to the number of hyponym links from t_e. We deal with this case by decreasing the type score in ranking candidate answer. In the second case, the entity is not a candidate answer. (As a simple example, consider t_q = *president*, $t_{e}e$ = *engineer*, t = *person*.)

2.3.3 Ranking Candidate Answer

We rank candidate answers using below formula.

$$score(Q,E) = type\_score(Q,E) * subtype\_score(Q_{subtype}, E_{subtype}) \\ *context\_score(Q_{context}, E)$$

In above formula, *type_score* is determined by the taxonomic relation between the answer type and the entity type and *subtype_score* is determined by the concept similarity between the subtype of question and subtype of entity. If the taxonomic relation between answer type and entity type corresponds to the first case in Figure 4, then *type_score* is 1, 1/2 for the third case, 0 for second case. *subtype_score* is calculated by below concept similarity formula.

$$Csim(C_i, P_j) = \frac{2 \times level(MSCA(C_i, P_j))}{level(C_i) + level(P_j)} \times weight$$

where $MSCA(C_i, P_j)$ is the most specific common ancestor of concept codes C_i and P_j. $level(C_i)$ is a level number on the Kadokawa thesaurus If C_i is a descendant of P_j, we set the *weight* to 1. Otherwise, we set the *weight* to 0.5.

context_score is calculated by term proximity value among other query terms occurred in the passage and the candidate answer as like following.

$$context\_score(Q,E) = \sum_i \frac{idf_i}{dist(q_i, E)}$$

where q_i is i-th query term, and *dist* is difference between document position of query term and document position of candidate answer.

We keep only single unique entity by removing redundancy entities. Two entities are redundant if their lexical words are the same.

Top N docs	Coverage	Precision
5 docs	0.4667	0.1703
10 docs	0.5179	0.1241
15 docs	0.5385	0.1002
20 docs	0.5538	0.0854
30 docs	0.5795	0.0656
100 docs	0.6513	0.0280
200 docs	0.6615	0.0158
500 docs	0.6769	0.0070
1000 docs	0.7026	0.0037

Table 3. Performance of document retrieval

Top N psgs	Coverage	Precision
5 psgs	0.3600	0.2200
10 psgs	0.4050	0.1505
15 psgs	0.4350	0.1230
20 psgs	0.4450	0.1018
30 psgs	0.4550	0.0757
100 psgs	0.5050	0.0286
200 psgs	0.5050	0.0143
500 psgs	0.5050	0.0057

Table 4. Performance of passage retrieval

3 Experiments

We participated in the QAC Task1 of NTCIR 4. Document collection consists of total 596,058 documents from 98-99 Mainichi and Yomiuri Newspaper articles.

3.1 Performance of Document Retrieval and Passage Retrieval

The performance of document retrieval and passage retrieval are shown in Table 3 and Table 4, respectively. Our document retrieval and passage retrieval shows weak performance. Among 197 questions, only 128 questions have the answer in top 100 documents, and only 98 questions in the top 100 passages. This retrieval results is different from NTCIR3 best system [13], although our retrieval methods is very similar to that system. The main reason for these results may be the difference of indexing units between two systems. Terms extracted by Chasen is not good for retrieval, and needs post-processing to make complex and reliable terms. For example, in Chasen, consecutive numeric value and bound noun are separated (for example, date), but these terms sometimes must be combined.

3.2 Performance of Answer Extraction

Question	Answer	Output	Correct
197	385	982	60
Recall	Precision	F-value	MRR
0.156	0.061	0.088	0.187

Table 5. Results of task 1 evaluation

Type	Ratio	# of questions	# of corrects	MRR
Person	24.36%	48	15	0.124
Org.	8.63%	17	4	0.088
Company	1.02%	2	0	0.000
Loc.	13.70%	27	10	0.131
Country	1.02%	2	1	0.083
Money	1.53%	3	2	0.233
Title	11.67%	23	3	0.052
Date	4.57%	9	3	0.105
Unit	10.15%	20	8	0.133
Other	23.35%	46	9	0.092
Total	**100%**	**197**	**55**	**0.187**

Table 6. Performance of answer extraction

Table 5 shows our results by the scoring tool (version 3.20). Table 6 shows the performance of answer extraction in our system for each answer type. The number of questions containing the correct answer in top 5 answer 55, and the coverage in top 5 is 27.91% (= 55/197). The pure performance of only answer extraction is 56.12% (= 55/98) if answer extraction uses top 100 passages. This performance is acceptable if we recognize that pure performance of answer extraction in NTCIR3 best system is 0.6917.

As shown in Table 6, the number of questions with other answer type is 48, occupying about 23% of total NTCIR-4 questions. However, for the other answer type, our system shows very weak performance (MRR: 0.092). The reason is that our named entity tagging focuses mainly to identify basic named entity class, neglecting other named entity types. In future, it will necessary to utilize automatic mechanism to construct named entities rules for extracting other class named entities.

4 Conclusion

Our QA system is the shallow-based. The main components of our system is question analysis and passage retrieval and named entity extraction. Passage retrieval was based on the density-based ranking method, which is the summation of idf values of query terms occurring in a passage. Named entity extraction was designed with the rule-based approach that uses lexico-semantic patterns, in which the Kadokawa thesaurus is a basic semantic resource. Although final performance of our QA system is low, the pure performance of answer extraction achieves 56.12%. In future, if we got high performed passage retrieval performance, our QA system could be a good system.

The experimental results showed that the method for passage retrieval is as critical as the answer extraction method. To increase passage retrieval, we will elaborate indexing unit and incorporate relevance feedback method of information retrieval into passage retrieval. In addition, we plan to design automatic methods to construct extraction rules for other named entity types.

References

[1] S. Abney, M. Collins, A. Singhal, "Answer Extraction", In Sixth Applied Natural Language Processing Conference, 2000.

[2] S. Buchholz. Using Grammatical Relations, Answer Frequencies and the World Wide Web for TREC Question Answering. The 10th Text REtrieval Conference, pages 502-509, 2002.

[3] S. Harabagui, D. Moldovan, M. Pasca, R. Mihalcea, M. Surdeanu, R. Bunescu, R. Girju, V. Rus, and P. Morarescu. Falcon: Boosting knowledge for answer engines. In the 9th Text REtrieval Conference, pages 479-488, 2000.

[4] S. Harabagui, D. Moldovan, M. Pasca, R. Mihalcea, M. Surdeanu, R. Bunescu, R. Girju, V. Rus and P. Morarescu. The Role of Lexico-Semantic Feedback in Open-Domain Textual Question Answering. In 39th Annual Meeting of the Asscociation of Computational Linguistics, pages 274-281, 2001.

[5] A. Ittycheriah, M. Franz, W. Zhu, A. Ratnaparkhi, IBM's Statistical Question Answering System. In 9th Text REtrieval Conference, pages 229-334, 2000.

[6] V. Keselj. Question Answering using Unification-based Grammar. Advanced in Artificial Intelligence, AI 2001, volume LNAI 2056 of Lecture Notes in Computer Science, 2001.

[7] D. Lin, P. Pantel. Discovery of Inference Rules for Question Answering. Natural Language Engineering, 7(4), pages 301-323, 2001.

[8] Y. Matsumoto, A. Kitauchi, T. Yamashita, Y. Hirano, H. Matsuda, and M. Asahara. Japanese morphological analysis system ChaSen version 2.0, 1999.

[9] A. Miheev, C. Grover, M. Moens. Description of the LTG System Used for MUC-7. In Proceedings of the MUC-7, 1998.

[10] S. Ohno, S. and M. Hamanishi. New Synonym Dictionary. Kadokawa Shoten, Tokyo, Japan, 1981.

[11] M. Pasca, S. Harabagui. High Performance Question/Answer. In 24rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pages 366-374, 2001.

[12] S.E. Robertson, S. Walker and M. Beaulieu. Okapi at TREC-7: automatic ad hoc, filtering, vlc and interactive. In the 7th Text REtrieval Conference, pages 253-264, 1999.

[13] Seungwoo Lee and Gary Geunbae Lee. SiteQ/J: A Question Answering System for Japanese. In Proceedings of the Third NTCIR Workshop, part IV. pages 31-38, 2002.
[14] M. M. Soubootin. Patterns of ponential answer Expressions as clues to the right answers. In the 10th Text REtrieval Conference (TREC-10), pages 293-302, 2001
[15] S. Tellex, B. Katz, J. Lin, A. Fernandes, G. Marton. Quantitative evaluation of passage retrieval algorithms for question answering. In the 26th Annual Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pages 41-47, 2003.
[16] J. Vicedo, A. Ferrandez. Importance of Pronominal Anaphora resolution in Question Answering systems. In Proceedings of 38th Annual Meeting of the Association for Computational Linguistics, pages 555-562, 2000.
[17] E. Voorhees, “The TREC question answering track”, Natural Language Engineering. 7(4), pages 361-378, 2001.