NTCIR-6 CLQA Question Answering Experiments at the Tokyo Institute of Technology

Josef Robert Novak Edward Whittaker Matthias Heie Shuichiro Imai Sadaoki Furui
Dept. of Computer Science
Tokyo Institute of Technology
2-12-1, Ookayama, Meguro-ku
Tokyo 152-8552 Japan
{novakj,edw,heie,imai15,furui}@furui.cs.titech.ac.jp

Abstract

In this paper we discuss our results from the 2006 NTCIR-6 CLQA task, subtasks 2a and 2b. We describe our language independent, data-driven approach to Japanese language question answering and our new document retrieval and answer projection method which resulted in a small performance gain in comparison to earlier approaches. Using this method, we achieve a formal run score of 0.17 for the top answer with document support for subtask 2b. We achieve a less favorable score of 0.03 for the top answer for the cross language subtask 2a, however we attribute this primarily to deficiencies in third-party MT software utilized for translation. We argue that these results further validate our current approach to QA.

Keywords: *NTCIR, Question Answering, Japanese, English, Non-linguistic, Answer Projection, Data-driven.*

1 Introduction

In this paper we discuss our results from the 2006 NTCIR-6 CLQA task, subtasks 2a and 2b, and briefly describe our language independent, data-driven approach to Japanese language question answering (QA). We also include a short comparison of our new Lucene-based IR system with our former Akechi-based approach.

The model we use for Japanese language QA is identical to the one we have now applied successfully to English language QA on the TREC tasks [10], and to English, French, and Spanish language QA on the CLEF tasks [13]. This model employs a novel statistical framework which is entirely data-driven and uses no morphological information as in for example [1, 7], or NE-tagging as in [1] and it does not perform any analysis of the question or of the target data as in [8]. Our system however, relies on some notion of a word as the basic modeling unit. Therefore we use Chasen 2.3.3 associated with the IPADIC 2.7.0 [6] dictionary but without ignore any of the morphological analysis that the system provides, for all segmentation related to training. Segmentation for IR however, was done in two different ways depending on the retrieval system and run, and is described below in Section 3.2.

Instead of linguistic information, our system is initially trained using n-gram statistics from a large example corpus of questions and corresponding answers (q-and-a). Answers to new questions are then extracted using statistical information obtained during the training process.

In the past we employed the Akechi system [2] for document retrieval, however this year we tested a new approach, based on the open source Lucene project [4] which achieved slightly better results. We outline this in Section 3.2.

Our top run for subtask 2b, (Japanese-Japanese), which achieved a score of 0.17 for the top answer with support, compares favorably with other submissions. Although our top run for subtask 2a, (English-Japanese), resulted in a less favorable score of 0.03, all other participants suffered similar relative drops in performance on the cross language tasks. In our case this was primarily due to using freely available web-based MT services to automatically translate English questions into Japanese.

The remainder of this paper is structured as follows. In Section 2 we outline our statistical classification approach to QA which is described more extensively in [10]. In Section 3 we describe the experimental setup and present the results obtained from NTCIR-6 CLQA task, subtasks 2a and 2b. In Section 4 we discuss the results and conclude in Section 5.

2 QA as Statistical Classification

The answer to a question depends on numerous different factors including the identities of the people involved, their immediate environmental and social con-

text, and previously asked questions. These and other similar contextual variables are clearly relevant in real-world situations, however they are very difficult to model and also to test in an off-line mode, such as that presented by NTCIR, CLEF and TREC evaluations. Therefore, we limit ourselves to modeling the most straightforward and obvious dependence: the probability of an answer A dependending on a question Q

$$P(A \mid Q) = P(A \mid W, X), \tag{1}$$

where A and Q are considered to be a string of l_A words $A = a_1, \ldots, a_{l_A}$ and l_Q words $Q = q_1, \ldots, q_{l_Q}$, respectively. Here $W = w_1, \ldots, w_{l_W}$ represents a set of features describing the "question-type" part of Q such as *when*, *why*, *how*, etc., while $X = x_1, \ldots, x_{l_X}$ represents a set of features that describe the "information-bearing" part of Q i.e. what the question is actually about and what it refers to. For example, in the questions, *Who is the oldest person in the world?* and *How old is the oldest person in the world?* the question-types *who* and *how old* differ, while the information-bearing component, *the oldest person in the world*, does not change.

Finding the best answer $\hat{A}$ involves a search over all available A for the one which maximises the probability of the above model i.e.,

$$\hat{A} = \arg\max_A P(A \mid W, X). \tag{2}$$

Given the correct probability distribution this is guaranteed to give us the optimal answer in a maximum likelihood sense. We don't know this distribution, and it is still difficult to model but, using Bayes' rule and making various simplifying, modeling and conditional independence assumptions (as described in detail in [10, 11, 12]) Equation (2) can be rearranged to give

$$\arg\max_A \underbrace{P(A \mid X)}_{\substack{retrieval \\ model}} \cdot \underbrace{P(W \mid A)}_{\substack{filter \\ model}}. \tag{3}$$

The $P(A \mid X)$ model is essentially a statistical language model that models the probability of an answer sequence A given a set of information-bearing features X. We call this model the *retrieval model* and do not examine it further (see [10, 11, 12] for more details).

The $P(W \mid A)$ model matches a potential answer A with features in the question-type set W. For example, it relates place names with *where*-type questions. In general, there are many valid and equiprobable A for a given W so this component can only re-rank candidate answers obtained by the retrieval model. We call this component the *filter model*, and it is structured as follows.

The question-type feature set $W = w_1, \ldots, w_{l_W}$ is constructed by extracting n-tuples ($n = 1, 2, \ldots$) such as *Where*, *In what* and *When were* from the input question Q. A set of $|V_{\mathcal{W}}| = 2522$ single-word features is extracted based on frequency of occurrence in our collection of example questions.

Modeling the complex relationship between W and A directly is non-trivial. We therefore introduce an intermediate variable representing classes of example questions-and-answers (q-and-a) c_e for $e = 1 \ldots |C_E|$ drawn from the set C_E. In order to construct these classes, given a set E of example q-and-a, we then define a mapping function $f : E \mapsto C_E$ which maps each example q-and-a t_j for $j = 1 \ldots |E|$ into a particular class $f(t_j) = e$. Thus each class c_e may be defined as the union of all component q-and-a features from each t_j satisfying $f(t_j) = e$. Finally, to facilitate modeling we say that W is conditionally independent of c_e given A so that,

$$P(W \mid A) \quad = \quad \sum_{e=1}^{|C_E|} P(W \mid c_W^e) P(c_A^e \mid A), \tag{4}$$

where c_W^e and c_A^e refer respectively to the subsets of question-type features and example answers for the class c_e.

The system using the model given by Equation (4) is referred to as model TWO. Model ONE which is described in [10] uses a slightly different derivation, however only systems based on model TWO were used in the official evaluations described in this paper.

3 Experimental Setup

In order to train the filter model for the system we use $|C_E| = 268,531$ example q-and-a from the 5TAKU quiz data [9] where each entry is composed of one question and five candidate answers e.g., **室町幕府の最後の将軍はだれ / 足利義昭, 徳川家茂, 徳川家康, 徳川慶喜, 徳川家斉**. Each class contains one unique question and one of its corresponding answers. We extract a set of $|V_{\mathcal{W}}| = 125$ single-word features from the most frequently occurring words in questions from the 5TAKU quiz data[1].

Finally, for the cross-language English-Japanese subtask, we rely on Google Translate [5] to automatically translate the English questions into Japanese. These translations are then fed as-is to our Japanese language QA system without any further processing.

3.1 Data sources

We use two different data sources for extracting answers: (1) the Mainichi Shimbun (1998-1999) newspaper corpus (`mai`) that was the official resource for both the NTCIR-3 QAC-1 task and this year's NTCIR-6 CLQA task, and (2) the top 300 Google documents corresponding to the question, which are downloaded

[1] This experimental setup is identical to that from [12].

Run	Retreival System	Top1
E-J-01	Akechi+mai	0.02
E-J-02	Akechi+mai+web	0.03
E-J-03	Lucene+mai	0.03
J-J-01	Akechi+mai	0.16
J-J-02	Akechi+mai+web	0.13
J-J-03	Lucene+mai	0.17

Table 1. Percentage correct answers in the Top1 position on six formal runs for NTCIR-6 CLQA track, subtasks 2a and 2b using 5000 mai documents.

at runtime (web). For each formal run one of mai, or mai+web was used.

3.2 Document retrieval

For document retrieval purposes we prepared two separate retrieval systems, Akechi-2.0.1b [2], and a modified version of the java-based open source text search engine library Lucene [4] which we equipped with a character-based segmenter. For each subtask we submitted one run each, using Akechi+mai, Akechi+mai+web, or Lucene+mai. Where Akechi+mai was combined with web data, Akechi was not used to index the web data, rather the method described in [10] was used to combine results from the two sources.

Our modified version of Lucene segments Japanese script into only character n-grams, numbers, and space delimited romaji words. The resulting segmented data is then indexed as unigrams, bigrams, and trigrams. For retrieval purposes questions are segmented according to the same rules and each n-gram is treated as an individual query term, and every document containing at least one query term is considered a "hit". Rank is determined by the sum of the individual tf*idf scores for all query terms found in a particular document.

3.3 NTCIR-6 CLQA formal runs

We participated in two subtasks for the NTCIR-6 CLQA task, the English-Japanese subtask 2a, and the Japanese-Japanese subtask 2b. We submitted three runs for each of these subtasks, resulting in a total of six formal run submissions. The individual runs for each subtask differed only in terms of the retrieval system used, as described above. The results for all six runs are displayed in Table 1 which shows the results from all six of our submissions. E-J-01, E-J-02, and E-J-03 represent the results for subtask 2a, while J-J-01, J-J-02, and J-J-03 represent the results for subtask 2b. The "Retrieval System" column specifies which retrieval system was used for a particular run.

4 Discussion

As can be seen in Table 1 we achieve our best score for the Japanese-Japanese task on run J-J-03, using the modified Lucene retrieval system with the mai corpus. This result is slightly better than the runs using either Akechi with the mai corpus, or Akechi+mai+web combination. Furthermore, although our best result on the English-Japanese subtask is considerably lower than any of our scores on the Japanese-Japanese subtask, our Lucene-based retrieval system still outperforms the Akechi only system. This performance difference on retrieval may be due in part to the fact that Akechi uses "word" based segmentation from Chasen, while our modified version of Lucene performs segmentation and indexing only on character n-grams, numbers and space delimited romaji words.

In order to confirm this hypothesis it would be necessary to further modify Lucene to perform word based segmentation, however there is some anecdotal evidence supporting the claim that character n-gram based indexing gives improved retrieval for longer queries [3].

It is interesting, and somewhat disappointing to note that the Akechi+mai+web combination achieved a lower score than Akechi+mai alone. In past experiments [10] we have never found a point at which performance deteriorates after a certain number of documents. Therefore, we suspect that the problem lies in the method used to combine the web and Akechi+mai results.

System performance on the Japanese-Japanese subtask, with a best run score of 0.17 for the top answer with support, was quite good. This score places us in the mid-range of all the participating systems, and also agrees favorably with those achieved on English in the TREC evaluations [10], and on English, Spanish and French in the CLEF evaluations [13].

Performance on the English-Japanese subtask, with a best run score of 0.03 for the top answer with support, was somewhat disappointing. This was due in large part to the decision to use freely available online web translation services to handle translation of the English questions into Japanese. No other additional processing was applied to the translated queries, and performance suffered as a result. This score also placed us at the bottom of the submissions for the English-Japanese cross language subtask.

Table 2 shows a breakdown by answer type, for correct answers on our best run on subtask 2b, run J-J-03. QType refers to the answer type for a given question, Qnum denotes the number of questions of this type included in the question set, 1st denotes the number of correct first-place answers for the question type, SErr refers to the number of answers marked wrong for segmentation errors, where an "answer segmentation error" refers to an inexact answer which either lacks necessary information or contains superfluous information.

It is easy to see that we obtain our best results on the "location" and "organization" types, while system

QType	Qnum	1st	SErr	Top1
Artifact	20	6	0	0.30
Date	31	5	5	0.16
Location	31	9	3	0.29
Money	13	1	2	0.08
Numex	20	3	0	0.15
Organization	20	5	0	0.25
Percent	15	1	2	0.08
Person	35	4	6	0.11
Time	15	0	0	0.00

Table 2. Typological breakdown for correct answers on subtask 2b, run J-J-03.

performance on number-like types ranges from 0.00 for "time" to 0.16 for "date". Low performance on the "time" type appears to be the result of time oriented questions being misinterpreted as "location" types. Of the 15 time-related questions such as, "**阪神大震災発生時刻は何時でしたか?**", twelve of the corresponding answers were "location" types with the answer to the preceding example given as "**兵庫県**". Misinterpretation of "time" type questions is most likely the result of a filter model error. Variation in system performance on other number-like types more likely reflects differences in the number of such questions in the test set, rather than anything specific to a particular type.

It is also interesting to note that, although performance on the "person" type was in the mid-range, there were six instances where a correct answer was found but marked wrong for a segmentation error. If these answers had been properly segmented, performance on the "person" category would increase to 0.29. Furthermore, if both unsupported answers and all segmentation errors are allowed, the overall top answer score for formal run J-J-03 rises to 0.28.

5 Conclusion

In this paper we have described our data-driven and non-linguistic approach to Japanese language QA, and also described our results on the NTCIR-6 CLQA subtasks 2a and 2b. We have now applied and tested this approach under evaluation conditions with English, Japanese, Spanish and French, and achieved roughly comparable accuracy levels in all languages. Our best run performance on the CLQA Japanese-Japanese subtask compared favorably with that of other participating systems, however performance on the cross language subtask was significantly worse.

The performance loss on the cross language subtask was largely due to the use of web based MT tools, thus in the future we may consider employing other techniques such as keyword lookup, on cross language exercises.

Our document retrieval system, based on the open source text search engine library Lucene, achieved a small improvement over the Akechi system we employed previously. This may be due to the character n-gram segmentation and indexing approach we used. In future we plan to test this hypothesis, and also to further investigate why the results which included web data were poorer than those obtained using only Mainichi shimbun data.

Finally, as mentioned our current approach is entirely data-driven, however this introduces some difficult problems with answer typing, as illustrated by our poor performance on "time" type questions. A demonstration of the system supporting English, Japanese, Spanish, French, Chinese, Russian and Swedish can be found online at `http://asked.jp/`.

6 Acknowledgments

This research was supported in part by JSPS and the Japanese government 21st century COE programme.

References

[1] M. Fuchigami, H. Ohnuma, and A. Ikeno. Oki QA System for QAC-2. In *Proc. of NTCIR-4 Workshop*, 2004.

[2] A. Fujii and K. Itou. Evaluating Speech-Driven IR in the NTCIR-3 Web Retrieval Task. In *Proc. of NTCIR-3 Workshop*, 2002.

[3] H. Fujii and B. Croft. A Comparison of Indexing Techniques for Japanese Text Retrieval. In *Proceedings of the 16th annual international ACM SIGIR conference on Research and Development in Information Retrieval*, 1993.

[4] `http://lucene.apache.org/`.

[5] `http://translate.google.com/`.

[6] Y. Matsumoto. User's Manual for Morphological Analysis system "Chasen" version 2.3.3. Technical report, NAIST, 2003.

[7] T. Mori. Japanese Q/A System using A* Search and Its Improvement. In *Proc. of NTCIR-4 Workshop*, 2004.

[8] S.-H. Na, I.-S. Kang, and J.-H. Lee. POSTECH Question-Answering Experiments at NTCIR-4 QAC. In *Proc. of NTCIR-4 Workshop*, 2004.

[9] Vector. Vector Software Library. In *Vector Software Library*, http://www.vector.co.jp/, 1995-2003.

[10] E. Whittaker, P. Chatain, S. Furui, and D. Klakow. TREC2005 Question Answering Experiments at Tokyo Institute of Technology. In *Proceedings of the 14th Text Retrieval Conference*, 2005.

[11] E. Whittaker, S. Furui, and D. Klakow. A Statistical Pattern Recognition Approach to Question Answering using Web Data. In *Proceedings of Cyberworlds*, 2005.

[12] E. Whittaker, J. Hamonic, and S. Furui. A Unifi ed Approach to Japanese and English Question Answering. In *Proceedings of NTCIR-5*, 2005.

[13] E. Whittaker, J. Novak, P. Chatain, P. Dixon, M. Heie, and Furui. CLEF2006 Question Answering Experiments at Tokyo Institute of Technology. In *Proceedings of the CLEF 2006 Conference*, 2006.