# A Unified Approach to Japanese and English Question Answering

Edward Whittaker Julien Hamonic Sadaoki Furui
Dept. of Computer Science
Tokyo Institute of Technology
2-12-1, Ookayama, Meguro-ku
Tokyo 152-8552 Japan
{edw,yuuki,furui}@furui.cs.titech.ac.jp

## Abstract

*We present our unified approach to question answering in different languages and describe our experiments on the Japanese language NTCIR-3 Question Answering Challenge (QAC-1) tasks 1 and 2. The model we use for Japanese language question answering (QA) is identical to the one we have applied successfully on the English language TREC QA tasks, based on a novel statistical, non-linguistic and data-driven approach to question answering. Using this method on the formal run of QAC-1 we obtain an MRR of 0.340 on task 1 and an average F-score of 0.159. The top1 accuracy of 26.5% compares very well with results obtained using an identical approach on the TREC evaluations.*

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

## 1 Introduction

In this paper we present our unified approach to question answering in different languages and describe our experiments on the Japanese language NTCIR-3 Question Answering Challenge (QAC-1) tasks 1 and 2. The model we use for Japanese language question answering (QA) is identical to the one we have applied successfully to English language QA on the TREC tasks [10]. Our QA system is based on a statistical, non-linguistic, data-driven approach to question answering, which uses the $N$-gram statistics from a large collection of example questions with corresponding answers (q-and-a) and large amounts of text data in which to find an answer. In contrast to other contemporary approaches to QA our English language system does not use WordNet [4, 6], named-entity (NE) extraction, or any other linguistic information e.g. from semantic analysis [4] or from question parsing [4, 5, 6] and uses capitalised word tokens as the only features for modelling. For our Japanese system, although we currently use Chasen to segment Japanese character sequences into units that resemble words, we make no use of any morphological information as used for example in [1, 7], we do not use NE-tagging as in [1] and we do not do any form of linguistic analysis of the question or of the data in which answers are sought [8]. Constructing QA systems from q-and-a data has already been proposed in [11] where a part-of-speech tagger is used to cluster questions. Our approach, however, uses only the word tokens in the question and answer strings during training and testing.

We show that with suitable training data, our approach can be applied effectively to two very different languages: Japanese and English. Moreover, each system achieves performance that is competitive with many contemporary systems though still somewhat worse than the best systems.

In Section 2 we give the highlights of our statistical classification approach to QA which is described more completely in [10]. In Section 3 we describe the experimental setup and present the results obtained on NTCIR-3 QAC-1 tasks 1 and 2. In Section 4 we discuss the results and conclude in Section 5.

## 2 QA as statistical classification

The answer to a question depends primarily on the question itself but also on many other factors such as the person asking the question, the location of the person, what questions the person has asked before, and so on. Although such factors are clearly relevant in a real-world scenario they are difficult to model and also to test in an off-line mode, for example, in the context of the NTCIR and TREC evaluations. We therefore choose to consider only the dependence of an answer $A$ on the question $Q$, where each is 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. In particular, we hypothesize that the answer $A$ depends on two sets of features $W = \mathcal{W}(Q)$ and $X = \mathcal{X}(Q)$ as follows:

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

where $W = w_1, \ldots, w_{l_W}$ can be thought of as a set of $l_W$ features describing the "question-type" part of $Q$ such as 誰 *(who)*, いつ *(when)*, どこ *(where)*, どん

な *(which)*, etc. and $X = x_1, \ldots, x_{l_X}$ is a set of $l_X$ features comprising the "information-bearing" part of $Q$ i.e. what the question is actually about and what it refers to. For example, in the questions, トム・クルーズはどこで結婚しましたか。*(Where was Tom Cruise married?)* and トム・クルーズはいつ結婚しましたか。*(When was Tom Cruise married?)* the information-bearing component is identical in both cases whereas the question-type component is different.

Finding the best answer $\hat{A}$ involves a search over all $A$ for the one which maximizes the probability of the above model:

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

This is guaranteed to give us the optimal answer in a maximum likelihood sense if the probability distribution is the correct one. Making various conditional independence assumptions as described in [10] to simplify modelling, we obtain the final optimisation criterion:

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

The $P(A \mid X)$ model is essentially a language model which models the probability of an answer sequence $A$ given a set of information-bearing features $X$. It models the proximity of $A$ to features in $X$. We call this model the *retrieval model* and examine it further in Section 2.1.

The $P(W \mid A)$ model matches an answer $A$ with features in the question-type set $W$. Roughly speaking this model relates ways of asking a question with classes of valid answers. For example, it associates dates, or days of the week with *when*-type questions. In general, there are many valid and equiprobable $A$ for a given $W$ so this component can only re-rank candidate answers retrieved by the retrieval model. If the filter model were perfect and the retrieval model were to assign the correct answer a higher probability than any other answers of the same type the correct answer should always be ranked first. Conversely, if an incorrect answer, in the same class of answers as the correct answer, is assigned a higher probability by the retrieval model we cannot recover from this error. Consequently, we call it the *filter model* and examine it further in Section 2.2.

## 2.1 Retrieval model

The retrieval model essentially models the proximity of $A$ to features in $X$. Since $A = a_1, \ldots, a_{l_A}$ we are actually modelling the distribution of multi-word sequences. This should be borne in mind in the following discussion whenever $A$ is used. As mentioned above, we currently use a deterministic information-feature mapping function $X = \mathcal{X}(Q)$. This mapping only generates word $m$-tuples ($m = 1, 2, \ldots$) from single words in $Q$ that are not present in an empirically built *stop-list* of around 50 high-frequency words for English and around 75 high-frequency words for Japanese. In principle the function could of course extract deeper linguistic features but we leave this for future work.

We first assume that a corpus of text data $S$ is available for searching for answers comprising $|S|$ sentences $S_1, \ldots, S_{|S|}$ and a set $U$ of $|U|$ documents and a vocabulary $V$ of $|V|$ unique words. We use the notation $X_i$ to define an active set of the features $x_1, \ldots, x_{l_X}$ such that $X_i = x_1 \cdot \delta(d_1), x_2 \cdot \delta(d_2), \ldots, x_{l_X} \cdot \delta(d_{l_X})$ where $\delta(\cdot)$ is a discrete indicator function which equals 1 if its argument evaluates true (i.e. its argument(s) are equal, is not an empty set, or is a positive number) and 0 if false (i.e. its argument(s) are not equal, is an empty set, is 0 or is a negative number) and $\vec{d} = [d_1, \ldots, d_{l_X}]$ is the solution to $i = \sum_{j=1}^{l_X} 2^{j-1} d_j$.

The probability $P(A \mid X)$ is modeled as a linear interpolation of the $2^{l_X}$ distributions:

$$P(A \mid X) = \sum_{i=1}^{2^{l_X}} \lambda_{X_i} \cdot P(A \mid X_i), \tag{4}$$

where $\lambda_{X_i} = 1/2^{l_X}$ for all $i$ and $P(A \mid X_i)$ is the conditional probability of $A$ given the feature set $X_i$ and is computed as the maximum likelihood estimate from the corpus $S$.

## 2.2 Filter model

A set of $|V_{\mathcal{W}}|$ single-word features is extracted based on frequency of occurrence in question data. Some examples include: いくら *(How much)*, どれ *(Which)*, どんな *(Which)*, 何 *(What)* etc. The question-type mapping function $\mathcal{W}(Q)$ extracts $n$-tuples ($n = 1, 2, \ldots$) of question-type features from the question $Q$, such as どの くらい *(How many)* and いつ まで *(Until when)*.

Modelling the complex relationship between $W$ and $A$ directly is non-trivial. We therefore introduce an intermediate variable representing classes of example q-and-a, $c_e$ for $e = 1 \ldots |C_E|$ drawn from the set $C_E$, and to facilitate modelling we say that $W$ is conditionally independent of $A$ given $c_e$ as follows:

$$P(W \mid A) = \sum_{e=1}^{|C_E|} P(W \mid c_e) \cdot P(c_e \mid A). \tag{5}$$

Given a set $E$ of example q-and-a $t_j$ for $j = 1 \ldots |E|$ where $t_j = (q_1^j, \ldots, q_{l_{Qj}}^j, a_1^j, \ldots, a_{l_{Aj}}^j)$ we define a mapping function $f : E \rightarrow C_E$ by $f(t_j) = e$. Each class $c_e = (w_1^e, \ldots, w_{l_{We}}^e, a_1^e, \ldots, a_{l_{Ae}}^e)$ is then obtained by $c_e = \bigcup_{j:f(t_j)=e} \mathcal{W}(t_j) \bigcup_{i=1}^{l_{Aj}} a_i^j$.

Assuming conditional independence of the answer words in class $c_e$ given $A$ and making the modelling assumption that the $j$th answer word $a_j^e$ in the example class $c_e$ is dependent only on the $j$th answer word in $A$ we obtain:

$$P(W \mid A) = \sum_{e=1}^{|C_E|} P(W \mid c_e) \cdot \prod_{j=1}^{l_{A^e}} P(a_j^e \mid a_j)$$
$$= \sum_{e=1}^{|C_E|} P(W \mid c_e) \prod_{j=1}^{l_{A^e}} \sum_{a=1}^{|C_A|} P(a_j^e \mid c_a) P(c_a \mid a_j), \quad (6)$$

where $c_a$ is a concrete class in the set of $|C_A|$ answer classes $C_A$, and assuming $a_j^e$ is conditionally independent of $a_j$ given $c_a$. The system using the above formulation of filter model given by Equation (6) is referred to as model ONE.

## 3 Experimental work

As mentioned earlier our system relies on some notion of words as the modelling units. We therefore use Chasen 2.3.3 associated with the IPADIC 2.7.0 dictionary for all question segmentation (both training, development and evaluation questions), answer segmentation and data segmentation. The morphological analysis output by Chasen with information such as part-of-speech, NE-tags etc. is not used in any way.

For training the filter model we use $|C_E| = 268531$ example q-and-a from the 5TAKU quiz data [9] where a question is posed together with 5 candidate answers such as: 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. Here, each class contains one unique example q-and-a. We remove any questions which overlap character-for-character with questions in the QAC-1 additional and formal runs. A set of $|V_{\mathcal{W}}| = 125$ single-word features is extracted from the most frequently occurring words in questions in the 5TAKU quiz data. The most frequent $|V_{C_A}| = 215000$ words from the Mainichi Newspaper (1998-1999) corpus were used to obtain $C_A$ for $|C_A| = 5000$ clusters.

### 3.1 Data sources

We use two different corpora as the source for locating answers to questions: (1) the Mainichi Shimbun (1998-1999) newspaper corpus (`mai`) that was the official source for the NTCIR-3 QAC-1 task; and (2) the NTCIR-3 WEB snapshot crawled in 2001 (`www`). For one set of experiments we also consider a combination of both the newspaper and web data (`mai+www`). Documents are retrieved using akechi-2.0.1b [2]. For each question and each data source, $|U| = 1, 5, 10, 50, 100, 500, 1000, 5000$ documents are retrieved. For `mai+www` we also use a combination of 5000 `mai` documents and 5000 `www` documents for each question.

### 3.2 Development: QAC-1 additional run

We use the 757 questions from the NTCIR-3 task1 additional run set for system development purposes. For determining system performance for tasks 1 and 2 we use the evaluation tool [3] provided after the NTCIR-3 QAC-1 conference. In addition we also compute the top1 accuracy as is now common in the NTCIR and TREC QA evaluations, however, we ignore the correctness of supporting documents in determining answer correctness.

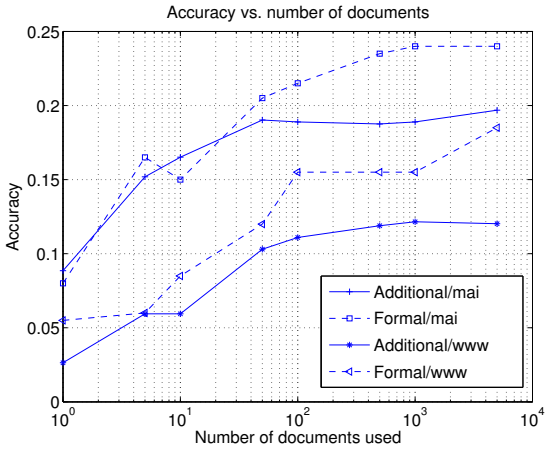


**Figure 1.** System accuracy vs. maximum documents used for NTCIR-3 QAC-1 additional and formal runs on 2 different data sources.

In Figure 1 the solid lines show the accuracy on the 757 additional run questions against the number of documents used for each of the two data sources. In the top half of Table 1 we show the best performing systems on the additional run using 5000 documents for each of `mai` and `www` and 10000 documents for the combined `mai+www` data source.

### 3.3 Evaluation: QAC-1 formal run

For the evaluation system we add in the 757 q-and-a from the NTCIR-3 task1 additional run set to the 268531 examples used during development. We then perform a final evaluation on the 200 questions from the NTCIR-3 QAC-1 formal run which was released before the additional run having previously ensured there was no overlap between them. The same scoring tool is used for assessing system performance.

In Figure 1 the dashed lines show the accuracy on the 200 formal run questions against the number of documents used for each of the two data sources. In the bottom half of Table 1 we show the results on the formal run obtained by the best performing systems on the additional run (given in the top half of Table 1).

| Run | Data | Accuracy | MRR | F-score |
|---|---|---|---|---|
| A | mai | 149 (0.197) | 0.260 | 0.131 |
| d | www | 91 (0.120) | 0.172 | 0.084 |
| d | mai+www | 154 (0.203) | 0.277 | 0.130 |
| F | mai | 48 (0.240) | 0.316 | 0.150 |
| o | www | 37 (0.185) | 0.237 | 0.106 |
| r | mai+www | 53 (0.265) | 0.340 | 0.159 |

**Table 1.** Accuracy, MRR and F-score on NTCIR-3 QAC-1 additional (Add) and formal (For) runs using 5000 mai documents and 5000 documents www and 10000 documents from mai+www.

## 4 Discussion

From the results in Table 1 we can see that the system performance is quite impressive despite the simplicity of our approach. While the performance is still somewhat lower than that of the best participating systems (MRR: 0.61 and F-score: 0.36) we are somewhere in the mid-range of all participating systems.

A particularly interesting observation is that the more data we use the better the results. While we have still not found an optimum (performance could conceivably deteriorate if too many documents are used) it appears from Figure 1 that performance is still increasing albeit at a slower rate. Moreover, using the 10000 documents from 2 different data sources gives us our best overall result with an MRR=0.34 on task 1 and an F-score=0.159 on task 2.

These results agree very favourably with those obtained on English in the TREC evaluations. In TREC2005 our model ONE system using only the supplied AQUAINT corpus achieved an official top1 accuracy of 14.3% when ignoring the need for correct document support. Our estimated performance of the model ONE system using web data instead was 17.7%. These results show that our model is equally effective for both English and Japanese language QA.

| % errors in each model combination | | | NOT |
|---|---|---|---|
| R | F | R&F | ERR. |
| 42.8% | 21.7% | 32.9% | 2.6% |

**Table 2.** Percentage of errors of total 152 in *R*etrieval, *F*ilter and *L*ength models, and *NOT* actually *ERR*ors best system on the QAC-1 formal run.

In Table 2 we give a subjective breakdown of which model is responsible for the errors on the best formal run. We see that the majority of errors are attributable to the retrieval model which was also the case for the English-language system [10] and reflects the simplicity of our current retrieval model. Improved models will therefore be investigated in the future. We also considered that 4 errors involving correct dates but without 年 were actually correct.

## 5 Conclusion

In this paper we have demonstrated the effectiveness of our unified approach to question answering on Japanese and shown that the performance is comparable with similar English language tasks. While the performance still falls short of the best systems around today, our system uses no linguistic information whatsover (except to perform character segmentation) instead relying on large quantities of real-world training examples and large amounts of data for searching for answers. In future we aim to complement our data-driven approach with a little more linguistic awareness and extend our unified approach to other languages.

A demonstration of the system using model ONE supporting questions in English, Japanese, 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] J. Fukumoto, T. Kato, and F. Masui. Question Answering Challenge (QAC-1) An Evaluation of Question Answering Task at NTCIR Workshop 3. In *Proc. of NTCIR-3 Workshop*, 2002.

[4] E. Hovy, U. Hermjakob, and L. C-Y. The Use of External Knowledge in Factoid QA. In *Proceedings of the TREC 2001 Conference*, 2001.

[5] A. Ittycheriah and S. Roukos. IBM's Statistical Question Answering System—TREC-11. In *Proceedings of the TREC 2002 Conference*, 2002.

[6] D. Moldovan, S. Harabagiu, R. Girju, P. Morarescu, F. Lacatusu, A. Novischi, A. Badulescu, and O. Bolohan. LCC Tools for Question Answering. In *Proceedings of the TREC 2002 Conference*, 2002.

[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. 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] L.-V. Lita, J. Carbonell. Instance-Based Question Answering: A Data Driven Approach. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing* , 2004.