Answering Yes-No Questions by Penalty Scoring in History Subjects of University Entrance Examinations

Yoshinobu Kano
Faculty of Informatics, Shizuoka University, Japan
kano@inf.shizuoka.ac.jp

Abstract

Answering yes–no questions is more difficult than simply retrieving ranked search results. To answer yes–no questions, especially when the correct answer is no, one must find an objectionable keyword that makes the question's answer no. Existing systems, such as factoid-based ones, cannot answer yes–no questions very well because of insufficient handling of such objectionable keywords. We suggest an algorithm that answers yes–no questions by assigning an importance to objectionable keywords. Concretely speaking, we suggest a penalized scoring method that finds and makes lower score for parts of documents that include such objectionable keywords. We check a keyword distribution for each part of a document such as a paragraph, calculating the keyword density as a basic score. Then we use an objectionable keyword penalty when a keyword does not appear in a target part but appears in other parts of the document. Our algorithm is robust for open domain problems because it requires no machine learning. We achieved 4.45 point better results in F1 scores than the best score of the NTCIR-10 RITE2 shared task, also obtained the best score in 2014 mock university examination challenge of the Todai Robot project.

1 Introduction

Although its importance has long been recognized (Hirschberg, 1984; Green et al., 1994), yes–no question answering (QA) has not been studied well compared to other types of QA such as factoid-style QA (Ravichandran et al., 2002; Bian et al., 2008) and non-factoid complex QA (Kelly et al., 2007), including definition QA (Cui et al., 2005; Xu et al., 2003).

As described herein, we propose an approach to answer yes–no questions. Our main claim is that it is necessary to handle *objectionable* keywords in *no* questions that are insufficiently considered in previous studies. We claim that this is the greatest difference in yes–no QA from other QA tasks. We suggest a penalized scoring method that finds and makes lower scores for objectionable keywords. This method can classify yes–no answers more sharply, overcoming the white noise effects described below.

In spite of the apparent simplicity that a yes–no question is a binary decision, it is not easy to answer. One might consider the following yes–no question.

(1) Is it dangerous to use an acidic cleaner with *enzyme* bleach?

A slightly different question can be posed by replacing *enzyme* with *chlorine*.

(2) Is it dangerous to use an acidic cleaner with *chlorine* bleach?

Example (1) includes the keywords *dangerous*, *acidic cleaner*, and *enzyme bleach*, while (2) includes *chlorine bleach* instead of *enzyme bleach*. Correct answers are *no* for (1) and *yes* for (2).

The standard means of answering yes–no questions would be to ask a search engine using keywords extracted as shown above. A search engine can return ranked results with confidence values. Comparing the topmost confidence values of yes and no questions, we can determine yes or no. However, standard search engines do not expect an objectionable keyword, *enzyme bleach* in (1). Therefore, they do not make a sufficient difference between (1) and (2), do not directly function for yes–no questions.

Yes–no QA can also be regarded as an application of factoid-style QA systems. In fact, (2) can be converted into the following.

(3) *What* is dangerous to use an acidic cleaner with?

Proceedings of the Open Knowledge Base and Question Answering (OKBQA) Workshop,
pages 91–96, Osaka, Japan, December 11 2016.

By replacing *chlorine bleach* with *What*, a factoid-style QA system (Mitamura et al., 2010) can provide an answer to question (3) such as *chlorine bleach*. By comparing the answer with the original question's keyword such as *chlorine bleach* in (2), *yes* or *no* can be assigned for each question (Prager et al., 2006). However, this conversion process includes a large part of the entire solution process as described below. The next example adds *in a washing machine* to (2), thereby producing the following question.

(4) Is it dangerous to use an acidic cleaner with chlorine bleach *in a washing machine*?

This addition does not affect the yes–no answer. When converting this question into a factoid-style question, which keyword to replace is a critical and difficult issue (Kanayama et al., 2012; Ishioroshi et al., 2014). The best system (Kobayashi et al., 2016) in the World History of the Todai Robot project's mock exam challenge combined different methods that make effective features unclear. These previous works leave some issues unresolved, what is the key feature to answer yes-no questions.

In either case, finding an objectionable keyword is the missing issue. Ideally speaking, all the keywords would co-occur in an evidence description of the knowledge source if the answer is *yes*. Unfortunately, keyword extraction is not perfect because it is extremely difficult to determine an unrelated keyword such as *washing machine*. Distribution of such an unrelated keyword has no relation to the co-occurrence of relevant and objectionable keywords. Consequently, it makes a sort of white noise in scoring. This effect produces a score difference between relevant and objectionable keywords vague. Standard frequency-based algorithms will not answer yes–no questions adequately.

Recognition of Textual Entailment (RTE) is another related task to the yes–no QA. RTE has recently been studied intensively, including shared tasks such as RTE tasks of PASCAL (Dagan et al., 2006; Giampiccolo et al., 2007), SemEval-2012 Cross-lingual Textual Entailment (CLTE) (Negri et al., 2012), and NTCIR RITE tasks (Kanayama et al., 2012). NTCIR-9 RITE (Shima et al., 2011) and NTCIR-10 RITE2's Exam Search tasks (Watanabe et al., 2013) required participants to find an evidence in source documents and to answer a given proposition according to yes or no. In this most realistic setting, no candidate sentence is given explicitly. One can consider the following, which is converted from question (1) of an interrogative form into an affirmative form.

(5) *It is* dangerous to use an acidic cleaner with enzyme bleach.

Judging entailment of (5) in a given source document is equivalent to answering yes–no question (1). Therefore, this style of RTEs can also be regarded as yes–no questions.

We describe details of our proposed method and implementation (Section 2), experiments and results (Section 3), discussion with potential future works (Section 4), and conclude the paper (Section 5).

2 Method and Implementation

Roughly speaking, our system performs (a) keyword extraction from the input, (b) keyword weighting of the input, and (c) source document search and scoring. Figure 1 shows our system architecture conceptually.

2.1 Keyword Extraction

We applied the same keyword extraction method both for the question text and the knowledge source text.

We performed an exact match in the given text for each page title of Wikipedia entries, and used matched titles as keywords. When exact match keywords overlap, we used only the longest match keyword, discarding shorter ones. Some page titles, such as single-letter words, were discarded manually to avoid illegal named entity matching. We regarded all page titles of Wikipedia's redirect pages as synonyms, i.e. identical words.

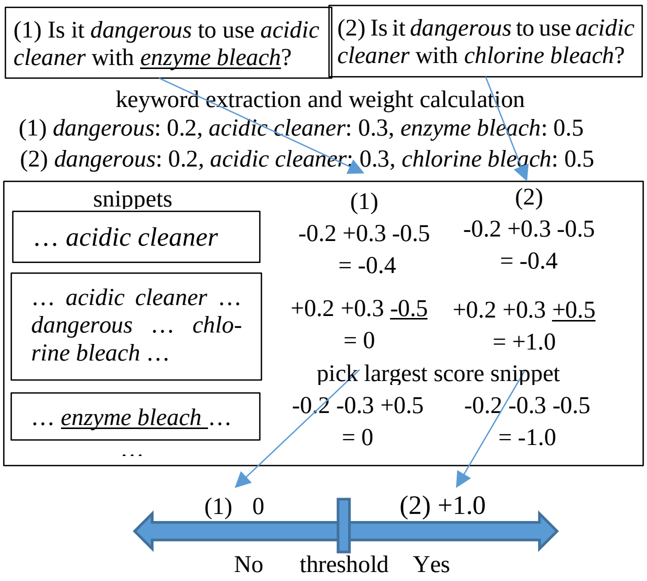
(1) Is it dangerous to use acidic cleaner with enzyme bleach?
(2) Is it dangerous to use acidic cleaner with chlorine bleach?
keyword extraction and weight calculation
(1) dangerous: 0.2, acidic cleaner: 0.3, enzyme bleach: 0.5
(2) dangerous: 0.2, acidic cleaner: 0.3, chlorine bleach: 0.5
snippets
(1)
(2)
… acidic cleaner
-0.2 +0.3 -0.5 = -0.4
-0.2 +0.3 -0.5 = -0.4
… acidic cleaner … dangerous … chlorine bleach …
+0.2 +0.3 -0.5 = 0
+0.2 +0.3 +0.5 = +1.0
pick largest score snippet
… enzyme bleach …
-0.2 -0.3 +0.5 = 0
-0.2 -0.3 -0.5 = -1.0
…
(1) 0
(2) +1.0
No
threshold
Yes

Figure 1. Conceptual figure of our system architecture.

Table 1. NTCIR-10 RITE2 Exam Search Results

	total #	proposed model						baseline	best in RITE2
source		Textbook			wikipedia			textbook	
snippet		sec	sub	p	sec	sub	p	p	
Y-F1	173	52.08	55.19	56.30	16.59	16.38	12.67	52.05	41.76
Y-Precision		47.39	52.33	60.69	10.98	32.20	29.17	49.48	57.00
Y-Recall		57.80	58.38	52.50	33.93	10.98	8.09	54.91	32.95
N-F1	275	64.06	69.06	68.83	71.36	70.78	71.41	67.04	74.48
N-Precision		69.20	71.76	72.58	60.71	60.41	60.25	69.53	66.67
N-Recall		59.64	66.55	65.45	86.55	85.45	87.64	64.73	84.36
Macro F1	448	58.07	62.12	**62.57**	43.98	43.58	42.04	59.55	58.12

Evaluation results in correct answer ratio of RITE2 official evaluation metric (b=3.2). source is knowledge source document. snippet is snippet unit: section (sec), subsection (sub), paragraph (p). Y/N-xx is correct answer yes/no.

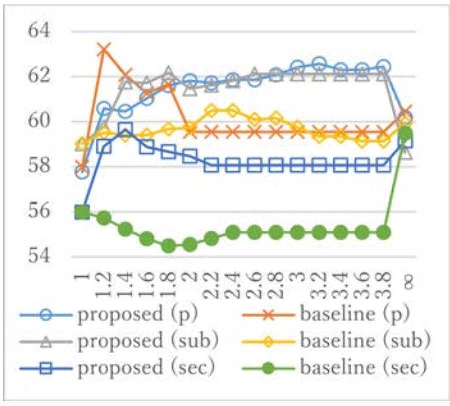

Figure 2. F1 scores w.r.t bias parameters.

2.2 Keyword Weighting

We assign a weight for each keyword that represents the importance of that keyword. Let c_i be the frequency of i-th distinct keyword in given knowledge source document. Then the weight of the i-th keyword is the following.

$$\mathrm{w}_i = 1/(\mathrm{c}_i z) + b$$

In this equation, $\mathrm{z} = \sum_i 1/c_i$ is a normalizing constant, where *i* is defined over the distinct keywords in the input. Also, *b* is a constant bias term that is optimized experimentally. A larger value of *b* decreases the effect of weight difference between keywords.

2.3 Document Search and Scoring

We assume that a relevant part of documents densely contains relevant keywords in a given question. This assumption is similar to most other existing methods.

We divide the source document data into snippets such as paragraphs. Snippets are manually predefined in our experiment knowledge source. We search for a snippet that has the highest score with respect to the input keyword set ***K***.

When a keyword such as *enzymatic bleaching* does not appear in a target snippet of the document, but appears in another snippet of the document, then we regard that keyword as objectionable with respect to the target snippet of the document and assign a lower score to the target snippet. This penalty enables us to construct a high-precision QA system using simple techniques. Let ***R*** be the keyword set extracted from a snippet. Then the score of ***R*** is

$$\mathrm{s}_{\boldsymbol{R}} = \sum_{l \in \boldsymbol{R} \cap \boldsymbol{K}} w_l - \sum_{m \in \boldsymbol{K} - \boldsymbol{R}} w_m$$

The first term of this expression means that the basic score of the snippet is the sum of the weights of the input keywords included in the snippet. The second term is a penalty term that subtracts the sum of the weights of the input keywords that are not included in the snippet, but included in another snippet. If a given choice is correct, then keywords in the choice should be included densely in a specific snippet of the source document. If a given choice is wrong, then its keywords should be scattered across snippets. The equation above penalizes such a scattered keyword distribution. Finally, we regard the maximum $\boldsymbol{s_R}$ among all snippets as the confidence score of the corresponding input. Yes–no is decided by comparison of the score with a threshold value, an average confidence score over a given dataset in our case.

We do not consider negations because it is rare for questions and source documents to describe events in a negative form.

3 Experiments and Results

The RITE2 Exam Search subtask was designed originally as an RTE task in which participants return true or false for a given proposition by referring to textual knowledge, such as Wikipedia and textbooks, with no candidate sentence in the knowledge source specified. The RITE2 dataset was developed from past Japanese National Center Test questions for the University Admissions (Center Test). The questions were presented originally in a multiple-choice style of questions. Because each choice corresponds to true or false, each choice can be regarded as a single yes–no question. Participant systems are asked to return yes or no with a confidence value for each question.

The dataset consists of a development set of 528 yes–no questions and a test set of 448 yes–no questions. All of our evaluation results are on the test set using the RITE2 official evaluation tool. Since our system requires no machine learning, we did not use the development set.

We used knowledge sources of two types: high school textbooks and Wikipedia. Both are written in Japanese. We tried three types of snippets: section, subsection, paragraph, larger to smaller in this order. Boundaries of these snippets are explicitly marked in textbooks by the textbook authors.

Wikipedia has its own document structures. For comparison with textbooks, we regarded a Wikipedia page as a section, sections in a page as subsections, and paragraphs as paragraphs. For efficiency, we used Wikipedia pages for which titles detected in the test datasets. This arrangement does not affect results because our keyword extraction is performed using the very same set of Wikipedia titles.

Table 1 shows results of *our proposed model, our baseline,* and the *best of RITE2* participant. The *source* row shows which knowledge source was used: either *textbook* or *Wikipedia*. The *snippet* row shows the snippet unit: *section*, *subsection,* or *paragraph*. Our baseline model is equivalent to the suggested model, except for dropping the penalty term, to check the effect of the penalty term. The baseline model becomes $s_{\boldsymbol{R}} = \sum_{l \in \boldsymbol{R} \cap \boldsymbol{K}} w_l$.

In the Macro F1 score, which was the primary metric in RITE2 balancing yes and no answers, our best system (knowledge source is *textbook* and snippet is *paragraph*) performed 5.45 points better than the best result in RITE2. Our best system performed 3.02 points better than our *baseline,* showing the effect of a penalty. Among the snippet units in our suggested method, *paragraph* using *textbook* obtained the best score overall. Results using *textbook* were better than those using *Wikipedia*. *Wikipedia* results do not show a clear difference irrespective of the snippet units.

Figure 2 shows a graph of the Macro F1 score with respect to the bias term *b*, with values of 1.0–3.8. The notation of ∞ is assigned when no weight is used. Comparison of pairs of *proposed* and *baseline* for each snippet shows that the *baseline* is almost always lower than *proposed*, i.e. the penalty term is effective. Table 1 corresponds to a bias value of *b* = 3.2.

4 Discussion

The result shows that our penalty scoring is effective in yes-no question answering.

Although we observed that keyword extraction was successful, keyword selection was difficult. A keyword that has no relation with the answer to the question could decrease the performance, even if our method is used.

The document structure granularity is another issue. Depending on a given question, a corresponding part of knowledge source differs. Its evidence might be described in a single sentence, or may be written using several sentences scattered across subsections. Our results imply that *paragraphs* are approximately the average size of the snippet per evidence description because *paragraphs* obtained the best score.

While result scores obtained using textbooks show a clear decreasing tendency when changing the snippet unit from smaller to larger, result scores obtained using Wikipedia are not clear. Write styles are different between textbooks' professional writers and Wikipedia's numerous anonymous writers. These differences are expected to produce various granularities in which part the evidence of a question we search for is described, producing the incoherent results. However, our results suggest that Wikipedia is still useful because of the word-based links, absorbing fluctuation of description and synonym variations.

A more difficult problem is the treatment of verbs. Noun synonyms can be covered well by the Wikipedia redirect relations and other existing dictionaries. However, finding relations between a pair of verbs is difficult. For example, to *suppress* someone and to *preserve* someone could be exclusive relations depending on their context; it would be difficult to produce such an exclusive word pair dictionary not just because it might depend on the context but also because the potential pairs are numerous.

While there is a couple of future work above, an advantage of our method is that no training is necessary when constructing the QA system. Another advantage is that we do not use any category of named entities. For these reasons, our system is domain-independent and robust for open-domain problems.

Our proposed method above is independent of any specific language. We can simply translate extracted keywords into the source document to perform cross-lingual searching if the given question is in a language (e.g. English) but not the same as a source document language (e.g. Japanese).

5 Conclusion and Future Work

We presented our method, which assigns importance to the objectionable keywords to answer yes–no questions accurately. We conducted experiments using the NTCIR-10 RITE2 shared task and others for comparison with previous studies. Results show that our system is a state-of-the-art system on the RITE2 task by 4.45 points better than the previous best system. The same system obtained the best score in World History of the mock examination challenge 2014 of the Todai Robot project. These results show that our penalty scoring is an effective feature to solve yes-no question answering.

Future work includes a better keyword selection depending on the context. A better scoring way using more precise document structure, and optimizing the yes–no threshold can also improve the results.

Reference

Bian, J., Liu, Y., Agichtein, E. and Zha, H. (2008). Finding the Right Facts in the Crowd: Factoid Question Answering over Social Media. *Proceedings of the 17th International Conference on World Wide Web*, WWW ’08 (pp. 467–476). inproceedings, New York, NY, USA: ACM. doi:10.1145/1367497.1367561

Cui, H., Kan, M.-Y. and Chua, T.-S. (2005). Generic Soft Pattern Models for Definitional Question Answering. *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR ’05 (pp. 384–391). inproceedings, New York, NY, USA: ACM. doi:10.1145/1076034.1076101

Dagan, I., Glickman, O. and Magnini, B. (2006). The PASCAL Recognising Textual Entailment Challenge. *Proceedings of the First International Conference on Machine Learning Challenges: Evaluating Predictive Uncertainty Visual Object Classification, and Recognizing Textual Entailment*, MLCW’05 (pp. 177–190). inproceedings, Berlin, Heidelberg: Springer-Verlag. doi:10.1007/11736790_9

Giampiccolo, D., Magnini, B., Dagan, I. and Dolan, B. (2007). The Third PASCAL Recognizing Textual Entailment Challenge. *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, RTE ’07 (pp. 1–9). inproceedings, Stroudsburg, PA, USA: Association for Computational Linguistics. Retrieved from http://dl.acm.org/citation.cfm?id=1654536.1654538

Green, N. and Carberry, S. (1994). Generating Indirect Answers to Yes-No Questions. *Proceedings of the Seventh International Workshop on Natural Language Generation*, INLG ’94 (pp. 189–198). inproceedings, Stroudsburg, PA, USA: Association for Computational Linguistics. Retrieved from http://dl.acm.org/citation.cfm?id=1641417.1641439

Hirschberg, J. (1984). Toward a Redefinition of Yes/No Questions. *Proceedings of the 10th International Conference on Computational Linguistics and 22Nd Annual Meeting on Association for Computational Linguistics*, ACL ’84 (pp. 48–51). inproceedings, Stroudsburg, PA, USA: Association for Computational Linguistics. doi:10.3115/980491.980503

Ishioroshi, M., Kano, Y. and Kando, N. (2014). A multiple-choice problem solver using question-answering system. *SIG Technical Reports, IPSJ-NL*.

Kanayama, H., Miyao, Y. and Prager, J. (2012). Answering Yes/No Questions via Question Inversion. *the 24th International Conference on Computational Linguistics (COLING 2012)* (pp. 1377–1391). Mumbai, India.

Kelly, D. and Lin, J. (2007). Overview of the TREC 2006 ciQA Task. *SIGIR Forum*, *41*(1), 107–116. article, New York, NY, USA: ACM. doi:10.1145/1273221.1273231

Kobayashi, M., Miyashita, H., Ishii, A. and Hoshino, C. (2016). NUL System at QA Lab-2 Task. *NTCIR-12 workshop* (pp. 413–420). Tokyo, Japan.

Mitamura, T., Shima, H., Sakai, T., Kando, N., Mori, T., Takeda, K., Lin, C.-Y., Song, R., Lin, C.-J., et al. (2010). Overview of the NTCIR-8 ACLIA Tasks: Advanced Cross-Lingual Information Access. *NTCIR-8 Workshop*.

Negri, M., Marchetti, A., Mehdad, Y., Bentivogli, L. and Giampiccolo, D. (2012). Semeval-2012 Task 8: Cross-lingual Textual Entailment for Content Synchronization. *Proceedings of the First Joint Conference on Lexical and Computational Semantics - Volume 1: Proceedings of the Main Conference and the Shared*

Task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation, SemEval ’12 (pp. 399–407). inproceedings, Stroudsburg, PA, USA: Association for Computational Linguistics. Retrieved from http://dl.acm.org/citation.cfm?id=2387636.2387700

Prager, J., Duboue, P. and Chu-Carroll, J. (2006). Improving QA Accuracy by Question Inversion. *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics*, ACL-44 (pp. 1073–1080). inproceedings, Stroudsburg, PA, USA: Association for Computational Linguistics. doi:10.3115/1220175.1220310

Ravichandran, D. and Hovy, E. (2002). Learning Surface Text Patterns for a Question Answering System. *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL ’02 (pp. 41–47). inproceedings, Stroudsburg, PA, USA: Association for Computational Linguistics. doi:10.3115/1073083.1073092

Shima, H., Kanayama, H., Lee, C., Lin, C., Mitamura, T., Miyao, Y., Shi, S. and Takeda, K. (2011). Overview of NTCIR-9 RITE: Recognizing Inference in TExt. *NTCIR-9 Workshop* (pp. 291–301). inproceedings, .

Watanabe, Y., Miyao, Y., Mizuno, J., Shibata, T., Kanayama, H., Lee, C.-W., Lin, C.-J., Shi, S., Mitamura, T., et al. (2013). Overview of the Recognizing Inference in Text (RITE-2) at NTCIR-10. *the NTCIR-10 Workshop* (pp. 385–404). Tokyo, Japan.

Xu, J., Licuanan, A. and Weischedel, R. M. (2003). TREC 2003 QA at BBN: Answering Definitional Questions. *TREC* (pp. 98–106). inproceedings, .