OKSAT at NTCIR-13 OpenLiveQ Task

- Mainly Offline Test Trials and Improvement -

Takashi SATO
Information Processing Center
Osaka Kyoiku University
Kashiwara Osaka Japan
+81-72-978-3823
sato@cc.osaka-kyoiku.ac.jp

ABSTRACT

Our group OKSAT submitted 21 runs for the NTCIR-13 OpenLiveQ task. We submitted from simple to complicate runs. Complicate runs are combinations of simple ones in most cases. We searched the question data mainly because we thought that the question data included the query string or related strings. We searched title, snippet and body by the query string, and merged their scores. We also took account page view and number of answers.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval – *Information filtering, Query formulation, Retrieval models, Search process, Selection Process.*

General Terms

Experimentation, Performance, Measurement.

Team Name

OKSAT

Subtasks

Keywords

Information Retrieval, Question-answering Service, Question Retrieval, Yahoo! Chiebukuro, Ambiguous Queries, Offline and Online Evaluation, Searching Title Snippet Body, Priority to Short Question Body, Page View, Number of Answers

1. INTRODUCTION

The construction of the QA system for ambiguous/ underspecified queries asked in Community Qestion Answering (cQA) services, is challenging problem. In order to evaluate the relevance in these systems, the relevance criteria should be changed from traditional one [1][2]. Our group OKSAT submitted 21 runs for the NTCIR-13 OpenLiveQ task. We submitted from simple to complicate runs. Complicate runs are combinations of simple ones in most cases. We searched the question data mainly because we thought that the question data included the query string or related strings. We searched title, snippet and body by the query string, and merged their scores. We also took account page view and number of answers

2. OUTLINE OF OUR APPROACH AND TARGET FIELD OF PROCESSING

We processed field variously which were extracted from the data provided by the task organizer. Figure 1 shows the outline of processing flow. We explain the name and the sign in the figure in this section and **3**.

We explain field which we used while referring to the figure as follows. From Question Data, we used the following field. The five boxes from the left of upper part of the figure.

We describe it in order of the field number, the notation in the figure and explanation in the task overview paper [2].

9: Page view; Page view of the question
8: Number of answers; Number of answers for the question
4: Title; Title of the question
5: Snippet; Snippet of the question in a search result
11: Body; Body of the question

In addition, it is not written in the figure, we used the following in one run(run2).

7: Update; Last update time of the question

We used one field, which is written rightmost in the figure, from Clickthrough Data in one run(run10).

4: Clickthrough rate; Clickthrough rate

3. PROCESSING ELEMENTS

We began the processing to make runs with the basic processing. Putting effective basic processing together, we made runs which required complicated processing. And we were adjusting parameters of the processing.

In this section, we explain the basic processing which is indicated by the sigh (P,A,T,S,K,B,L,M,C) circled and the box contacted with in Figure 1. In the next section, we explain concretely how to made runs assembling these basic processing using the sighs.

P: Maps the Page View expressed with an integer onto the number of 0-1. We call it normalization in order to merge with another score.
A: Similar to P, we normalized the Number of Answers.
T: About the number of searched words to search Title by Question string, we calculated score of the Title in probabilistic model based on Tf-ifd (simplified Okapi BM25) [3].
S: Similar to T, we calculated score of the Snippet.
K: About the length of Snippet, we made the threshold and calculated score by a calculating formula to give priority to a short one over.
B: Similar to T, we calculated score of the Body.
L: Similar to K, we calculated score about the length of Body.
M: We performed morphological analysis of the Query, and made plural search words from each query string which could be divided.

C: Similar to P, we normalized the Clickthrough.

In addition, it is not written in the figure, but there are the following three basic processing.

N: We extracted nouns by morphological analysis [4][5] of the title and snippet.

U: Case-insensitive search.

Z: Full and half size insensitive search.

4. HOW TO MAKE RUN

Using the notation of the target field of processing in **2** and the basic processing of **3**, we show how to make runs which we submitted. We attach the combination of basic processing notation of **3** surrounded by [and] in the following run's title. Table 1 shows the evaluation result (nDCG@10) of offline test for submitted runs.

Table 1. Evaluation results of offline test

run	nDCG@10	run	nDCG@10
run0	0.35451	run11	0.33449
run1	0.37083	run12	0.37958
run2	0.29214	run13	0.41960
run3	0.29426	run14	0.24125
run4	0.36388	run15	0.42514
run5	0.30756	run16	0.40094
run6	0.32638	run17	0.43241
run7	0.30427	run18	0.43516
run8	0.33365	run19	0.43767
run9	0.37837	run20	0.44471
run10	0.36669		

4.1 No Processing

run0

Nothing done from Question data. We simply extracted Query ID and Question ID from the top to the lower row of Question data. By the task overview [2], Question data is the output of top 1,000 questions retrieved from Yahoo! Chiebukuro by each question.

4.2 Single Processing

In this subsection, we explain runs which have single basic processing in **3**.

run1 [P]

We sorted the questions in the Question data by the number of the page view of their question.

run2 [U]

We sorted by the last update time (Update in **2**) of the questions in the Question data. Newer questions are ranked higher. nDCG@10 is not so good. As last update time of the data is mostly 2016 year and near, the newer one is not so important in this case.

run3 [L]

We sorted questions by the length of the body (Body in **2**) of each question in the Question data. The longer questions were ranked higher.

run4 [L]

Inverse order of run3. In other words, the shorter questions were ranked higher. The nDCG@10 is higher than run3. However we thought too short Body is not good, we set threshold length in the next run (run5).

run5 [L]

Setting 300 byte (100 characters of Japanese full-width character in utf-8 code) as threshold of the length of the body, We made the reciprocal number of the square root of the ratio of the length as score. The nDCG@10 was lower than run4, so we made run7 later.

run6 [B]

We counted the number of times included in the Body for each query string.

run7 [L]

This is the same as run5 except that the threshold of the text length becomes 150byte (300 byte for run5).

run14 [N]

We calculated tf-idf of each noun which was extracted by morphological analysis of the title and snippet, and then we added them.

4.3 Simple Combination of Processing

In this subsection, we explain runs which have the processing P and/or L plus at most one other processing. We also add one run (run11) which has the similar processing only.

run8 [B,L]

We divided the number of times of the string included in Body by the square root of length of Body.

run9 [P,L]

We merged the effect of run1 and run7. We divided the Page view by the square root of length of the Body. We set the threshold of the length of Body for 100byte.

run10 [P,L,C]

We merged the effect of run9 and clickthough rate of Clickthrough data. Click through data are available for the restricted questions though. We did not use Clickthrough after this run because nDCG@10 of this run is lower than run9.

run11 [T,S,B]

As well as Body, we counted the number of times including the query string about Title and Snippet. We normalized these three numbers from 0 to 1, and then we summed them.

4.4 Complex Combination of Processing

In this subsection, we explain runs which have complex combination of processing of **3**. The combination is briefly noted in the title of each run.

run12 [P,T,S,B]

We normalized Page view and then we added the score of run11.

run13 [P,T,S,B,L]

We divided score of run12 by the square root of length of Body. The threshold length of the Body was set to 100byte.

run15 [P,T,S,B,L,U]

This is the same as run13 except that the case insensitive string matches were done.

run16 [P,T,S,B,L,U,Z]
This is the same as run15 except that we converted full size alphanumeric characters into half size alphanumeric characters.

run17 [P,T,S,B,L,U,Z]
The files were handled in binary until run15, but from run16 files were handled in utf-8. So, we set the threshold of the body length to 30 characters (about 1/3 of 100 byte).

run18 [P,T,S,B,L,U,Z,M]
When as a result of having performed morphological analysis of the query string, it was divided into plural words, we searched the Title, the Snippet and the Body by those words also.

run19 [P,T,S,B,L,U,Z,M,A]
We normalized the Number of answers and then we added the score of run18.

run20 [P,T,S,B,L,U,Z,M,A,K]
We set threshold of the Snippet length to 200, then we add reciprocal number of the cubic root of the ratio of the Snippet length to run19.

5. OFFLINE and ONLINE TEST

Online test is assessed for the top run of each participation group by multileaving [6] method. Our top run of offline test was the best run of all participants, however, it was not good under online test. We imagined that the taste of the judgment of online test was different from offline test. We felt that it was difficult to show the question list which the user expected without having the information about the taste of the user. So, the profile of the user may help to improve the performance of cQA systems if possible.

6. CONCLUSIONS

Our group OKSAT submitted 21 runs for the NTCIR-13 OpenLiveQ task. We submitted from simple to complicate runs. Complicate runs are combinations of simple ones in most cases. We searched the question data mainly because we thought that the question data included the query string or related strings. We searched title, snippet and body by the query string, and merged their scores. We also took account page view and number of answers. Our top run of offline test was the best run of all participant, however, online test of our top run of offline test was not good. It was difficult to show the question list which the user expected without having the information about the taste of the user. So, the profile of the user may help to improve the performance of cQA systems if possible.

7. ACKNOWLEDGMENTS

Our thanks to Yahoo Japan Corporation for allowing us to use the Yahoo! Chiebukuro data.

8. REFERENCES

[1] M. P. Kato and Y. Liu, Overview of NTCIR-13 , in *Proceedings of the NTCIR-13 Conference*, Tokyo, Japan, 2017.

[2] M. P. Kato, T. Yamamoto, T. Manabe, A. Nishida and S. Fujita, Overview of NTCIR-13 OpenLiveQ Task, in *Proceedings of the NTCIR-13 Conference*, Tokyo, Japan, 2017.

[3] S.E. Robertson and S. Walker, Some simple effective approximations to the 2-Poisson model for probabilistic weighted retrieval, in *Proceedings of the 17th International Conference Research and Development in Information Retrieval*, pp. 232-241, 1994.

[4] MeCab: Yet Another Part-of-Speech and Morphological Analyzer, http://taku910.github.io/mecab/ (2017/10/01).

[5] mecab-ipadic-NEologd : Neologism dictionary for MeCab, https://github.com/neologd/mecab-ipadic-neologd/blob/master/README.ja.md (2017/10/01).

[6] A Schuth, F Sietsma, S Whiteson, D Lefortier and M de Rijke, Multileaved comparisons for fast online evaluation, in CIKM, pp. 71–80, 2014.

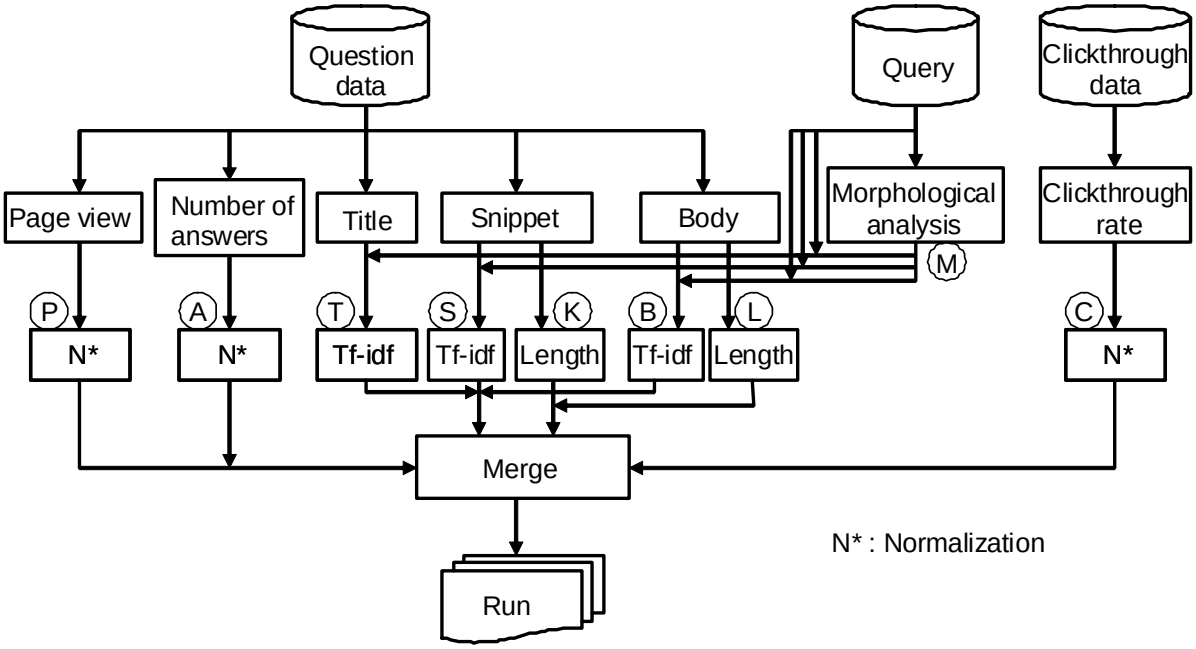

Figure 1. Outline of processing flow