An Effective Question Expanding Method for Question Classification in cQA Services

Kyoungman Bae and Youngjoong Ko
Department of Computer Engineering, Dong-A University
840, Hadan 2-dong, Saha-gu, Busan, 604-714, Republic of Korea
kyoungman.bae@gmail.com, youngjoong.ko@gmail.com

ABSTRACT
This paper introduces a new question expanding method for question classification in cQA services. Input questions are mostly generated by a small size of text in the cQA services, and test inputs consist of only a question whereas training data do a pair of question and answer. Thus, the input questions cannot provide enough information for good classification in many cases. To solve this problem, we propose the question expanding method by pseudo relevant feedback and automatic answer generation. For pseudo relevant feedback, we first find relevant question-answer pairs related to an input question using the Indri search engine, and then top relevant words are chosen as expanded words. The automatic answer generation tries to create pseudo answers by adding question-related words using translation probabilities from questions to answers by Giza++. As a result, we obtain the significant improved performances when two approaches are effectively combined.

Categories and Subject Descriptors
H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval; I.2.7 [Natural Language Processing]: Text analysis

Keywords
Question Classification, cQA Service, Pseudo Relevant Feedback, Question Expanding, Translation Probability

1. INTRODUCTION
A community Question Answering (cQA) service has used a question classification technique to recommend the candidate categories when a user enters his/her question into the cQA system. Question classification in the cQA service is quite different from traditional text classification and factoid question classification from the TREC QA [1, 2].

- The focus of question classification in Question Answering (QA) systems is to predict the type of the answer. On the other hand, the focus of question classification in cQA services is to classify and recommend the category or topic of a question. It is strongly associated with the text classification.

The input questions in the cQA services consist of a small size of text, different from long texts in the traditional text classification. Thus, they commonly suffer from insufficient information from the short size of questions to well classify the topic of the questions.

To solve the above challenges, we propose a new question expanding method in the cQA services for question classification. To effectively select expanded words, we focus on two approaches: the pseudo relevant feedback and the generating pseudo answer. For the former, pseudo relevant feedback has traditionally used for query expansion in information retrieval systems [3] and it can be also applied to question classification in the cQA services. Relevant question-answer pairs to the input question are selected as top ranked question-answer pairs from the Indri search engine [4] and then words in relevant question-answer pairs are ranked according to estimated important scores by reflecting the ratio of word occurrences in relevant and non-relevant question-answer pair sets. Finally, top ranked words are added to an original question as expanded words. For latter, the training data of the cQA services commonly has more information than input test question because each training example is composed of a pair of a question and its answer but test example is only a question. We assume that the answer is strongly associated with the input question. Based on this assumption, we try to create a pseudo answer to expand each input question. For this, we exploit translation probabilities from questions to answers in the training corpus using the Giza++ toolkit. The words with high translation probabilities, which are estimated in the training corpus by Giza++, are added into the pseudo answer. The added words are limited only words that occur in relevant question-answer pairs from the pseudo relevant feedback and this helps us obtain more accurate expanded words. A pair of an original input question and its pseudo answer is regarded as an input to the cQA services. Finally, we try to effectively combine the expanded word list and the created pseudo answer for generating a final expanded input text to the cQA services.

We evaluated the proposed method by comparing the baseline models. In our experiment, the proposed model achieved 89.7%, which is 6.8% higher performance than baselines. As a result, the proposed method improved the performance of question classification significantly.

This paper is organized as follows. In section 2, our question expanding method for question classification is described in detail. In section 3, we discuss the analysis of experimental results and discussion. Finally, we draw some conclusions in section 4.
2. RELATED WORK

A variety of researches in cQA service are focused on the question retrieval. And the idea of finding similar questions in cQA using the translation probabilities for words based on similar question pairs has been proposed previously [5, 6]. Compared with the previous studies, we first focus on classifying the categories of questions in cQA. We think it can contribute to the improvement of the cQA system because it effectively provides limited categories for each question.

Question classification in TREC QA has been studied during the past decade. It is to assign one or more categories to a given question written in natural language and the set of predefined categories are usually called question taxonomy or answer type taxonomy, such as “Location,” “Human,” and so on. Since typical Question Answering (QA) systems automatically generate the answer, they use the question classification to detect the entity type of the answer that is the question taxonomy [1]. On the other hand, the focus of question classification in cQA services is to classify and recommend the category of the question because the answer in cQA services is directly generated by user.

Traditionally, various text classification techniques have been studied [7, 8, 9]. They present several statistical methods for text classification on two kinds of textual data, such as newspaper articles and e-mails. In fact, question classification of the cQA service has a different aspect from traditional text classification; the input of question classification in the cQA service is short because it is a question. L. Cai et al. [2] proposed a question classification method in cQA. It focuses on the large-scale question classification. And, they use the Wikipedia semantic knowledge for expanding questions. We also use the expanding method for the question classification based on the pseudo relevant feedback and the translation probability. A lot of methods are proposed for query expansion based on the pseudo relevant feedback [10, 11]. In query expansion, selecting good expansion words is very important [12]. For this, the relevant score is used [13, 14]. In this paper, we use the translation score using the translation probability between questions and answers for ranking the expansion words.

3. PROPOSED METHOD

One of the major challenges for question classification is to effectively expand input questions. In particular, application areas just like a cQA service, which has an input text with smaller number of words, require more effective question expanding methods. Therefore, we propose to apply pseudo relevant feedback and automatic answer generation techniques to question answering for the cQA services.

Our strategy is processed through the following steps:

Step 1: given an input question, a list of top 20 ranked question-answer pairs is chosen from the Indri search engine as relevant question-answer pairs.

Step 2: all nouns in the relevant pairs are considered as candidate words for question expansion. Then we rank the candidate words by combining calculated relevant scores from pseudo relevant feedback and the translation scores from Giza++. Finally, we can obtain top n ranked words as expanded words.

Step 3: we enlarge the input question by adding the expanded words into original question.

3.1 Extracting the candidate words for expanding

When using the Indri search engine, we can obtain the top-ranked relevant question-answer pairs for each input question. Then all nouns, which occur in more than one relevant question-answer pair among top 20 question-answer pairs, are regarded as the candidate words for expanding an input question. To choose expanded words with good quality, we exploit translation scores as well as relevant scores from pseudo relevant feedback.

3.2 Generating expanded words

The top ranked question-answer pairs commonly include some non-relevant question-answer pairs to an input question. They make noisy information if we use whole of candidate words. To select the good quality of expansion words, we first sort the all candidate words according to combination scores of the calculated relevant scores and the estimated translation scores. Then the high-ranked candidate words are selected as expanded words.

3.2.1 Calculating the relevant scores by pseudo relevant feedback

The relevant score is calculated using Robertson and Walker’s relevance weights [13, 14].

\[ RW(Q, w) = r_w \log \frac{N}{n_w} - \log \frac{R}{r_w} - \log V \]  

(1)

where \( Q \) is an input question, \( RW(Q, w) \) is the relevance score of word \( w \), \( R \) is the total number of top-ranked relevant question-answer pairs, \( r_w \) is the number of top-ranked relevant question-answer pairs in which the word \( w \) occurs, \( N \) is the size of the collection, \( n_w \) the number of question-answer pairs that contain the word \( w \), and \( V \) is the size of the vocabulary. The argument of the second logarithm is the number of ways one can choose \( r_w \) from \( R \). We set \( R \) to 20.

3.2.2 Calculating the translation probabilities

The translation scores are calculated using the translation probabilities, which are estimated by Giza++. The question-answer pairs are considered as a type of parallel corpus that is used for estimating word-to-word translation probabilities [3]. In this approach, we estimate the word-to-word translation probabilities by regarding questions as a source and answers as a target.

\[ TW(Q, w) = \frac{1}{|Q|} \sum_{qw \in Q} P_{\text{trans}}(w|qw) \]  

(2)

where \( P_{\text{trans}}(w|qw) \) is the translation probabilities between two words \( w \) and \( qw \), estimated by the Giza++ toolkit. \( TW(Q, w) \) is the mean probability between a candidate word, \( w \), and question words in input question. \( Q \) denotes the length of input question.
Combining the relevant scores and the translation scores
To improve the performance, we try to merge two scores: the relevant score and the translation score. We first normalize two scores divided by maximum score individually. Each score ranges from 0 to 1. Then we calculate combination scores by linearly combining two scores and normalizing with the maximum score as follows:

\[
CW(Q, w) = \alpha NRW(Q, w) + \beta NTW(Q, w)
\]

\[
NCW(Q, w) = \frac{CW(Q, w)}{\max(CW(Q, t_i), i = 1, ..., n)}
\]

\[
NRW(Q, w) = \frac{RW(Q, w)}{\max(RW(Q, t_i), i = 1, ..., n)}
\]

\[
NTW(Q, w) = \frac{TW(Q, w)}{\max(TW(Q, t_i), i = 1, ..., n)}
\]

where \(CW(Q, w)\) is the combined score, \(NRW(Q, w)\), \(NTW(Q, w)\) and \(NCW(Q, w)\) are normalized scores, \(n\) is the number of the candidate words related \(Q\), \(t_i\) is a candidate word. \(\alpha\) and \(\beta\) are parameters for linear combination. We set \(\alpha\) to 0.2 and \(\beta\) to 0.8 in our experiment.

3.3 Adding expanded words into an original input question
Finally, top \(n\) words ranked by combination scores in equation (7) are chosen as expanded words and they are added into original input question as follows:

\[
\bar{Q}_{\text{expand}} = \alpha \bar{Q}_{\text{original}} + \beta \bar{E}\bar{W}
\]

where \(\bar{Q}_{\text{original}}\) is an original input question vector and \(\bar{E}\bar{W}\) is an expanded words vector. \(\alpha\) and \(\beta\) are also parameters for linear combination of the original words in an input question and expanded words. We set \(\alpha\) to 1 and \(\beta\) to 0.75. We use the TFIDF score as the weight of each word in the vector. The expanded words have an additional weight, such as the relevant weight, the translation weight and the combination weight. When expanding the words, we use this weight as importance of each expanded word by multiplying the TFIDF.

\[
\bar{E}\bar{W} = \bar{E}\bar{W}_{\text{tfidf}} \times \bar{E}\bar{W}_{\text{importance}}
\]

where \(\bar{E}\bar{W}_{\text{tfidf}}\) is an expanded words vector, which has the TFIDF as the weight. \(\bar{E}\bar{W}_{\text{importance}}\) is also an expanded words vector, which has the importance score as the weight.

4. EXPERIMENTS

4.1 Data Sets and Experimental Settings
To verify the proposed method, we collected a total of 14,702 question-answer pairs from the best Q&A group in Naver KiN. Among original 10 categories on Naver KiN, we used all categories (Computer Communication, Game, Travel, Shopping, Sport, etc.). Note that the size of question-answer pairs in Naver KiN remarkably varies in the number of the content words. The distribution of our corpus is as follows:

As shown in Figure 1, the 91% of content words are the noun. We think that the noun is more important than other types of the content words, such as verb, adjective, etc.

The 60% of questions have 20 or fewer noun words.

For evaluation, we used the five-fold cross-validation. As an evaluation measure for the performance of category classification, we computed the F1 measure.

\[
F1 = \frac{2PR}{P + R}
\]

SVM was selected as a state-of-the-art classifier and the TFIDF scheme was applied to word weighting for all the classifiers. We constructed the multi-category classifier using SVM classifiers. Every category has a classifier like Figure 3. Classifier, indicates the classifier of category \(c_k\) and Score\(_k\) is the prediction score of classifier\(_k\).
4.2 Baseline Systems
The baseline systems in our experiments are defined as follows:

**Table 1. Train and Test in the baseline systems**

<table>
<thead>
<tr>
<th></th>
<th>Baseline</th>
<th>GST</th>
</tr>
</thead>
<tbody>
<tr>
<td>Train</td>
<td>Question &amp; Answer</td>
<td>Question &amp; Answer</td>
</tr>
<tr>
<td>Test</td>
<td>Question</td>
<td>Question &amp; Answer</td>
</tr>
</tbody>
</table>

**Baseline**: a classifier is trained using the questions and related answers. And then, the classifier is tested using only test question.

**Gold Standard Test (GST)**: a classifier is trained using questions and their answers. And then, test questions and their golden standard answers are equally used in a test phase. In practice, an input test question does not have its answer. However, we can use answers associated with test questions because the test questions are randomly selected from the training corpus. It is called the Gold Standard Test.

**Table 2. Performance comparison of the baseline systems (%)**

<table>
<thead>
<tr>
<th></th>
<th>Baseline</th>
<th>GST</th>
</tr>
</thead>
<tbody>
<tr>
<td>micro-avg.</td>
<td>82.9</td>
<td>87.4</td>
</tr>
<tr>
<td>macro-avg.</td>
<td>82.6</td>
<td>87.1</td>
</tr>
</tbody>
</table>

where micro-avg. denotes a micro-averaging F1 score and macro-avg. a macro-average F1 score. As shown in Table 2, GST achieved better performance than Baseline because it uses golden standard answers.

**Table 3. Performance comparison of the baseline systems using the other types of the content words**

<table>
<thead>
<tr>
<th></th>
<th>micro-avg.</th>
<th>macro-avg.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Noun</td>
<td>82.9%</td>
<td>82.6%</td>
</tr>
<tr>
<td>Noun &amp; Adjective</td>
<td>83.1%</td>
<td>82.9%</td>
</tr>
<tr>
<td>Noun &amp; Verb</td>
<td>83.0%</td>
<td>82.7%</td>
</tr>
<tr>
<td>Noun &amp; Verb &amp; Adjective</td>
<td>83.2%</td>
<td>82.9%</td>
</tr>
<tr>
<td>All</td>
<td>82.5%</td>
<td>82.1%</td>
</tr>
</tbody>
</table>

As shown in Table 3, since the performances using the other types of the content words are similar to the performance using the noun content words, we evaluated our model using only noun content words.

4.3 Question expanding methods
We here evaluate proposed question expanding methods using the relevant scores, NRW, the translation scores, NTW and combined scores, NCW. Table 4 shows a performance comparison when input questions are expanded according to three different word ranking based on NRW, NTW and NCW.

**Table 4. Performance comparison of three word ranking approaches**

<table>
<thead>
<tr>
<th></th>
<th>Baseline</th>
<th>NTW</th>
<th>NRW</th>
<th>NCW</th>
</tr>
</thead>
<tbody>
<tr>
<td>micro-avg.</td>
<td>82.9</td>
<td>84.7(+1.8)</td>
<td>89.4(+6.5)</td>
<td>89.7(+6.8)</td>
</tr>
<tr>
<td>macro-avg.</td>
<td>82.6</td>
<td>84.2(+1.6)</td>
<td>89.1(+6.5)</td>
<td>89.4(+6.8)</td>
</tr>
</tbody>
</table>

As a result, the question expanding method based on NCW scores achieved the best F1 score of 89.7%. The top 10 ranked words were selected as the best expanded words.

5. CONCLUSIONS AND FUTURE WORK
This paper has studied a new question expanding method for question classification in a cQA service. Pseudo relevant feedback and automatic answer generation were exploited to choose expanded words. Relevant question-answer pairs from the Indri search engine and translation probabilities from Giza++ were utilized in our proposed method. As a result, when we effectively combine the relevant scores from pseudo relevant feedback and the translation scores from translation scores, we achieved the best F1 score of 89.7%. It is 6.8% higher performance than one of baseline.

In future work, we plan to apply the proposed expanding method to question retrieval for improving the performance of the cQA service.

6. ACKNOWLEDGMENTS
This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (No. NRF-2013R1A1A2009937)

7. REFERENCES


