Exploring Social Q&A Collections for Answering Complex Questions

Youzheng Wu **Hideki Kashioka**
Spoken Language Communication Group, Keihanna Institute, NiCT
3-5 Hikari-dai, Seika-cho, Soraku-gun, Kyoto, Japan
(youzheng.wu,hideki.kashioka)@nict.go.jp

Abstract

This paper investigates techniques to automatically collect training data from social Q&A collections such as Yahoo! Answer for supporting sentence-based complex question answering (QA) system. Using the collected training data, we construct a targeted-answer-style classifier for each type of questions and adopt it to remove non-targeted-answer-style sentences before using any state-of-art IR formula to select answers. Experiments on the 10 types of Chinese complex questions show that our system can significantly outperform the baseline.

1 Introduction

In the study of QA systems, several types of questions such as factoid, definition, reason and opinion questions have been studied. Among the approaches proposed to answer these questions, machine learning based techniques are more effective in constructing QA components from scratch. However, these supervised techniques require a large scale of <question, answer> pairs (Q&A pairs) as training data: e.g., [Echihabi et al., 2003] and [Sasaki et al., 2005] constructed 90,000 English Q&A pairs, and 2,000 Japanese Q&A pairs, respectively for their factoid QA systems. [Biadsy et al., 2008] constructed 53,426 biographical & non-biographical training sentences for their definition QA system. [Higashinaka et al., 2008] used 4,849 positive & 521,177 negative examples for their reason QA system.

Along with the principle of the supervised machine learning techniques, we have to reconstruct training Q&A pairs for new types of questions such as hazard-type questions (What're the hazards of popped food), impact-type questions (List the impact of the financial turmoil on Southeast Asia.), etc, which is too expensive and labor intensive. To deal with the acquisition problem of training Q&A pair data, this paper resorts to social Q&A collections crawled from the Web. The proposed sentence-based complex QA system consists of: 1) Automatically collecting training Q&A pairs from the crawled social Q&A collection for each type of questions; 2) Extracting valuable features to train a targeted-answer-style classifier for each type of questions; 3) Removing non-targeted-answer-style sentences before ranking sentences for answer selection. We evaluate our system in terms of 10 types of Chinese questions with an adaption of evaluation tool Pourpre V.0c [Lin et al., 2006], which shows that our system is effective.

2 Social Q&A Collections

Recently, some new-style social QA websites such as Yahoo! Answer, Baidu Zhidao, etc, appear on the Web, which provide an interactive platform for users to post questions and answers. The Q&A pairs included in such communities increase dramatically, which could be a source of training data required in supervised machine learning based QA systems. In this paper, we are interested in exploring such user-generated Q&A collections for building Q&A training data.

The Q&A collections have two salient characteristics: textual mismatch between questions and answers, i.e., question words are not necessarily repeated in answers; and user-generated spam or flippant answers, which are the unfavorable factors for our study. Therefore, we just crawl those Q&A pairs, which questions have best answers tagged by users. Finally, 6.0 million Q&A pairs are crawled from Chinese social QA websites, which is employed as the source of training data in our study.

3 Our Complex QA System

Conventional complex QA system is a cascade of the following modules: Question Analyze: analyzing test questions and identifying answer types of questions. Document Retrieve & Candidate Answer Generate: retrieving relevant documents to questions from the given collection (1998-2001 Xinhua and Lianhe Zaobao Newspapers used in this paper) for consideration, and segmenting the documents into sentences. Answer Selection can apply any state-of-art IR formulas (i.e., the KL-divergence language model) to estimate similarities between sentences (1024 sentences used in our case) and questions, and select most similar sentences as final answers. To calculate similarities, the Answer Selection mainly explores features such as textual similarities, keyword density and frequency, Web correlation between question and its answers.

We argue that it is possibly helpful to classify the extracted sentences into targeted-answer-style sentences and non-targeted-answer-style sentences by using type-of-question-dependent properties such as targeted answer word format, part-of-speech (PoS) format, etc., and then select final answers from targeted-answer-style sentences. We, thereby, adapt the above architecture via applying a targeted-answer-style filter before the Answer Selection module. Accordingly, the problems that remain are: constructing targeted-answer-style and non-targeted-answer-style training data, and training classifiers using features extracted from the data collected.

3.1 Collecting Training Data

We first introduce the notion of answer type informer of the question as: a short sub-sequence of tokens (typically 1-3 words) in question that are adequate for question classification; e.g.: *hazard* in question of *what are the hazards of global warming?* This paper regards answer type informer recognition as a sequence tagging problem and adopts conditional random fields (CRF). We labeled 3,262 questions with answer type informer manually to train a CRF, which classifies each question word into a set of tags O = {I_B, I_I, I_O}: I_B for a word that begins an informer, I_I for a word that occurs in the middle of an informer, and I_O for a word that is outside of informer. In the following feature templates, w_n and t_n refer to word and PoS, respectively; n refers to the relative position from the current word n=0. The feature templates include: w_n and t_n where n=-2,-1,0,1,2; w_nw_{n+1} and t_nt_{n+1} where n = -1,0; $w_nw_{n+1}w_{n+2}$ and $t_nt_{n-1}t_{n-2}$ where n = -2,-1,0; and O_nO_{n+1} where n=-1, 0.

According to the informers of questions identified by the trained CRF, we can cluster Q&A pairs that have the same informers as target-answer-style training data of the corresponding type of question, e.g.: the Q&A pairs grouped via informer *hazard* are regarded as target-answer-style training data of answering hazard-type questions. For each type of question, we randomly select some Q&A pairs that do not contain informers in questions as non-targeted-answer-style training data. Table 1 reports the number of the target-answer-style training QA pairs obtained for each type of test questions. The preprocessing of the training data includes word segmentation, PoS tagging, NE tagging [Wu et al., 2005]. We also replace each NE by its tag type.

3.2 Classifiers

We extract lexical-based, PoS-based n-grams as features from the target-answer-style training data to train classifiers. To reduce the dimensionality of feature space, we first select top 3,000 lexical unigrams using $score_w$ = tf(w) * log(idf_w), where, tf(w) denotes the frequency of word w, idf(w) is the inverted document frequency of w that indicates its global importance. To learn lexical bigrams and trigrams, top 300 unigrams are used as seeds, and the iteration procedure is shown in algorithm 1.

Algorithm 1: Extracting Lexical-based n-grams

1: $k \leftarrow 2$
2: ***while*** S^{k-1} is not empty
3: ***for*** each s^{k-1} in S^{k-1} ***do***
4: Add a preceding or following word of S^{k-1} to form k-gram, s^k;
5: ***if*** s^k exists in S^k ***then*** $s^k_{.freq}$++;
6: ***else*** $s^k_{.freq} \leftarrow 1$; $S^k \leftarrow s^k$;
7: ***for*** each s^k in S^k ***do***
8: ***if*** $s^k_{.freq} > \theta 1$ ***then*** $S^k \leftarrow s^k$
9: k++

To learn the PoS-based features, we adapt the algorithm 1 by using all part-of-speeches as seeds, and replacing line 4 with "Add a proceeding or following PoS of s^{k-1} to form k-gram, s^k". Moreover, we assign each extracted feature s_i with a weight calculated by $c_1^{s_i} / (c_1^{s_i} + c_2^{s_i})$, where $c_1^{s_i}$ and $c_2^{s_i}$ denote its fre-

quencies in targeted-answer-style and non-targeted-answer-style training data, respectively.

As classifier, we use linear classification SVMs, which can directly optimize multivariate performance measures [Joachims et al., 2005]. We held out 90% of training QA pairs for training classifiers, and 10% of them for test. The (Precision, F-Measure) scores of the SVM classifiers with optimizing error-rate (percentage of errors in predictions), and prec@k (precision of a classifier that predicts exactly k = 100 examples to be positive) on the held-out test data are (84.2%, 57.9%), and (78.9%, 64.9%), respectively. These results indicate that the classifiers can achieve good precision, but F-measure is not so satisfactory when classifying social-generated Q&A pairs. We will validate these results for our complex QA system in experimental section.

4 Experiments

As far as we know, there is no standard data set for evaluating complex QA system. We, therefore, created data set ourselves, which consists of 10 types of Chinese complex questions, i.e.,危害/hazard-type,影响/impact-type, 态度/attitude-type, 意义/significance-type,事件/event-type,作用/function-type,原因/reason-type,措施/treatment-type,伤亡/casualty-type, and 规模/scale-type questions. Table 1 shows the statistic of the test data. For each test question, we also provide a list of weighted answer nuggets. Evaluation is conducted via the adaption of Pourpre v1.0c [Lin et al., 2006] that uses the standard scoring methodology for TREC other questions, i.e., answer Nugget Recall NR, Nugget Precision NP, and an combination score F3 of NR and NP. For better understanding, we evaluate the systems when outputting top N sentences as answers.

qtype	#[1]	#[2]	qtype	#[1]	#[2]
hazard	10	10,362	function	5	41,005
impact	10	35,097	significance	10	14,615
attitude	10	1,801	treatment	5	3,643
reason	10	10,241	casualties	7	102
event	15	3,260	scale	5	642

Table 1: Numbers of test questions (#[1]) and the training QA pairs learned(#[2])

Table 2 reports the evaluation results for several N values. The baseline refers to the conventional method introduced in Section 3, which does not employ target-answer-style filter before answer selection. This experiment shows that: 1. incorporating targeted-answer-style filter can greatly outperform the baseline, and the advantage of our systems becomes more obvious with the increasing of N; 2. $\text{Ours}_{\text{error-rate}}$ is better than $\text{Ours}_{\text{pre@k}}$ when N is less than 10, this is because the precision of the classifier optimizing error-rate is superior to the classifier optimizing prec@k.

Figure 1 exhibits how well the $\text{ours}_{\text{pre@k}}$ system performs for each type of questions when N is set to 10. This figure indicates that our method improves the performance of the baseline on all types of test questions. The largest improvement, 20%, is from casualty-type questions, which is due to high performance (100% of precision, and 88.9% of F-measure) of the classifier on casualty-type questions. The absolute enhancements in terms of treatment-type, scale-type and significance-type questions are smallest, which are 1.8%, 1.9%, and 2.9%, respectively.

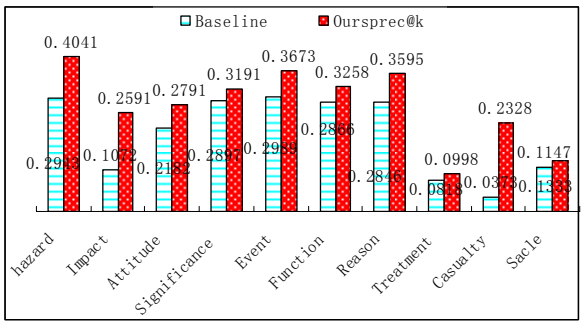

Figure 1: F3 performance by type of questions.

5 Related Work

Recently, many studies have been done on such social-generated Q&A collections. [Surdeanu et al., 2008] proposed an answer ranking engine to rank answers to non-factoid English questions in the Q&A collection. [Duan et al., 2008] proposed a MDL-based tree cut model to search similar English questions that have been answered. Therefore, these studies just search answers from social QA websites. While our study focuses on mining knowledge from social Q&A websites for automatic QA system. This paper is similar to [Mori et al., 2008], which extracted Bi-grams from Japanese Q&A collection to improve their automatic QA system.

	F3 (%)			NR (%)			NP (%)		
	N=1	*N=5*	*N=10*	*N=1*	*N=5*	*N=10*	*N=1*	*N=5*	*N=10*
Baseline	9.82	18.18	21.95	9.44	19.85	27.84	34.35	25.32	18.96
$Our_{error\text{-}rate}$	11.88	23.09	27.10	11.62	26.54	36.61	32.89	26.32	16.76
$Ours_{prec@k}$	10.94	22.60	30.05	10.61	25.52	39.98	30.41	26.88	18.56

Table 2: Overall performance for the test data

6 Conclusion

This paper investigates the technique of exploring social Q&A collections to acquire training data for supervised machine learning based complex QA system, which is proved to be effective. Removing noise Q&A pairs in the training data will be conducted in future work.

References

Ryuichiro Higashinaka and Hideki Isozaki. Corpus-based Question Answering for why-Questions. In Proc. of IJCNLP 2008, pp.418-425.

Yutaka Sasaki. Question Answering as Question-biased Term Extraction: A New Approach toward Multilingual QA. In Proc. of ACL 2005, pp215-222.

Abdessamad Echihabi and Daniel Marcu. A Noisy-Channel Approach to Question Answering. In Proc. of the ACL 2003, Japan.

Mihai Surdeanu, Massimiliano Ciaramita, and etc. Learning to Rank Answers on Large Online QA Collections. In Proc. of ACL 2008, pp719-727.

Fadi Biadsy, Julia Hirschberg, and Elena Filatova. An Unsupervised Approach to Biography Production using Wikipedia. In Proc. of ACL 2008, pp807-815.

Ves Stoyanov, Claire Cardie, and Janyce Wiebe. Multi-Perspective Question Answering Using the OpQA Corpus. In Proc. of HLT/EMNLP2005, pp923-930.

Youzheng Wu, Jun Zhao, Bo Xu, and Hao Yu. Chinese Named Entity Recognition Model based on Multiple Features. In Proc. of HLT/EMNLP 2005, pp427-434.

Huizhong Duan, Yunbo Cao, Chin Yew Lin, and Yong Yu. Searching Questions by Identifying Question Topic and Question Focus. In Proc. of ACL 2008, pp156-164.

Tatsunori Mori, Takuya Okubo, and Madoka Ishioroshi. A QA system that can answer any class of Japanese non-factoid questions and its application to CCLQA EN-JA task. In Proc. of NTCIR2008, pp41-48.

Jimmy Lin and Dina Demner-Fushman. Will Pyramids Built of Nuggets Topple Over? In Proc. HLT/NAACL2006, pp 383-390.

Thorsten Joachims. A Support Vector Method for Multivariate Performance Measures. In Proc. ICML2005, pp 383-390.