ECNU at TREC 2015: LiveQA Track

Weiqian Zhang, Weijie An, Jinchao Ma, Yan Yang, Qinmin Hu and Liang He
Shanghai Key Laboratory of Multidimensional Information Processing
East China Normal University, 500 Dongchuan Road, Shanghai, 200241, China
{wqzhang, wjan, jcma}@ica.stc.sh.cn, {yanyang, qmhu, lhe}@cs.ecnu.edu.cn

Abstract

This paper reports on East Normal China University's participation in the TREC 2015 LiveQA track. An overview is presented to introduce our community question answer system and discuss the technologies. This year, the Trec LiveQA track expands the traditional QA track, focusing on "live" question answering for the real-user questions. At this challenge, we built a real-time community question answer system. Our results are presented at the end of the paper.

1 Introduction

The automated question answering (QA) track, which has been one of the most popular tracks in TREC for many years, focuses on the task of providing automatic answers for human questions. The track primarily deals with factual questions, and the answers provided by participants are extracted from a corpus of News articles. While the task evolves to model increasingly realistic information needs, addressing question series, list questions, and even interactive feedback, a major limitation remains: the questions do not directly come from real users in real time.

The LiveQA track revives and expands the QA track, focusing on "live" question answering for real-user questions this year. Real user questions, extracted from the stream of most recent questions submitted on the Yahoo Answers (YA) site that have not yet been answered by humans, will be sent to our systems. Then our system provides an answer in real time.

This paper introduces our question answering system, which we use for extracting answers for real-user questions in real time. Since many questions submitted on these CQA sites like Yahoo Answers, have been asked by someone else or have been answered somewhere else on the web, we assume that most the real-user question can be solved if we can make full use of the existing question-answer pairs on the Web. In this task, we convert the focus from answering the given question to finding the best answer in the existing problems-answer pairs, where we make use of the existing question-answer pairs as the training data.

2 System Overview

Our system contains three parts: the QA search module, the question selection module and the answer selection module. Figure 1 illustrates the architecture.

In the QA search module, the questions first analyzed, comprising the full and shallow parses and the named entity tagging module. We used these semantic information to expand the original questions to our new queries. We apply these new queries to search all relevant question-answer

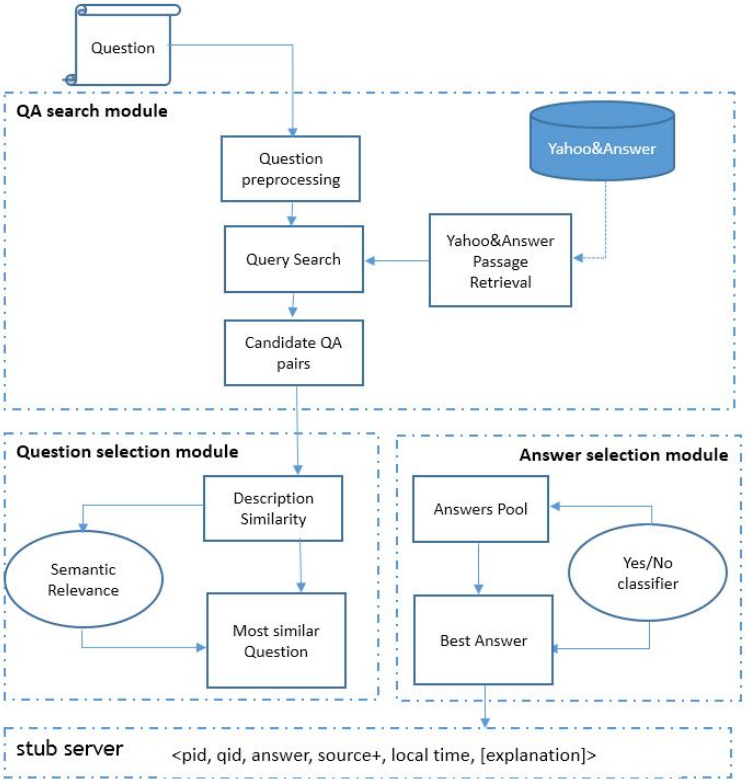

Figure 1: Relevance Prediction Framework

pages on Yahoo Answers as the candidates. Then, we utilize our question selection module to find out the most similar question and choose the best answer given this question.

3 Approaches

3.1 The QA search module

After analyzing the Yahoo Answers site, we find out that the search service provided by Yahoo Answers is appropriate for the QA search. By using the Web service, users can search the questions and sort the answers by the relevance, the waiting time or the number of answers. In Table 1, we evaluate our experimental results configuring multiple strategies.

We use the sample of 1000 Yahoo Answers QIDs[1] in the experiment and manually input 100 questions of them as our queries. The results returned for each query are judged on a YES/NO scale. Note that "YES" means that at least one question has the same meaning with the query or contains a suitable answer for our query in the candidate results.

Table 1: The number of YES for each strategy.

Strategy	Relevance	Time	Number of answers
Number of YES	64	53	58

We finally choose the relevance strategy for our system to search questions on Yahoo Answers and develop a spider module to generate candidate questions automatically.

3.2 The Question Selection Module

By studying the candidates generated by the QA search module, we find that Yahoo sorted the questions in terms of the semantic similarity between the query and the candidate question title. However, Yahoo did not use the semantic information from the question description. Then, we add the semantic information in this module to re-calculate the relevance between the query and each candidate question using LSI and LDA[3]. LSA and LDA is a widely used corpus-based measure when evaluating textual similarity. We adopt the vector space sentence similarity, which represents each sentence as a single distributional vector and sums up the LSA/LDA vector of each word in the sentence. After that, the original rank sorted by Yahoo is integrated with the similarity as candidate.

3.3 The Answer Selection Module

In this module, we choose the best answer among the candidate question. First, we treat this task as a supervised classification problem[4]. First, our training data contain the L6 corpus[2] and 190 thousand question pages from Yahoo Answer in categories specified by the organizers. Then, we calculate the probability of each answers as the best answer, through applying the logistic regression.

Furthermore, we extract features[4] from multiple sources of CQA-based information, i.e. bag-of-words and answer-specific features from answer, string matching and semantic similarity from QA pair, question specific features from question. Specifically, to measure the semantic similarity between two documents, we employ two WordNet-based word similarity metrics: Unt[1] and LCH[2] similarity. We also use the LSI/LDA method mentioned above in question selection module to calculate the text similarity.

[1] https://github.com/yuvalpinter/LiveQAServerDemo/blob/scraping/data/1k-qids.txt

[2] http://webscope.sandbox.yahoo.com/catalog.php?datatype=l

4 Results

Our system answers all the 1083 questions provided by the LiveQA server, in which 331 questions are returned the default answer as "yes". The specific metrics include: the average answer score, precision (fraction of answered questions with a score above a threshold), and coverage (fraction of all questions answered).

Overall, 1087 questions have been measured and scored using 4-level scale. The performance measures are:

- avg-score(0-3) - average score over all queries (transferring 1-4 level scores to 0-3, hence comparing 1-level score with no-answer score, also considering -2-level score as 0)
- succ@i+ - number of questions with i+ score (i=1..4) divided by number of all questions
- prec@i+ - number of questions with i+ score (i=2..4) divided by number of answered only questions

Table 2 shows the scores of the answers corresponding to the 1087 questions. We get higher score in all the performance evaluation. Especially, we get thirty percent higher in the succ@4 and prec@4 than the average score, that means our system has good performance in answering the questions that has been asked before.We can find out the most relevant question and its specific answer.

Table 2: The evaluation results.

	avg score (0-3)	succ@1+	succ@2+	succ@3+	succ@4+	prec@2+	prec@3+	prec@4+
ECNU_ica	0.567	0.971	0.289	0.191	0.089	0.297	0.197	0.092
Average	0.465	0.925	0.262	0.146	0.060	0.284	0.159	0.065

5 Conclusion

We have presented our method and experiments result in solving the LiveQA task. And we get good performances in solving the questions come from the real users, in real time.

When processing the nature language questions, we intend to understand the deep meaning that the user really want to know. It is hard to use a machine-learning method to achieve that. And we found that it is difficult to answer the real user questions(mostly not the factual questions) using a state of the art QA technology.

In the future, we are planing to do more research on the LiveQA question. Considering the superiority of IR technology, we intend to combine our system with advanced retrieval system to solve the opinion questions. To dealing with the factual questions, we can refer to the QA task before.

Acknowledgment

This work is supported by the Science and Technology Commission of Shanghai Municipality under research grant No.14511107000 and No.14511106803.

References

[1] C. Banea, S. Hassan, M. Mohler, and R. Mihalcea. Unt: A supervised synergistic approach to semantic text similarity. In *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, pages 635–642. Association for Computational Linguistics, 2012.

[2] C. Leacock and M. Chodorow. Combining local context and wordnet similarity for word sense identification. *WordNet: An electronic lexical database*, 49(2):265–283, 1998.

[3] C. Wang, L. Cui, B. Yang, and X. Wu. Question recommendation mechanism under q&a community based on lda model. *Open Cybernetics & Systemics Journal*, 8:645–650, 2014.

[4] L. Yi, J. Wang, and M. Lan. Ecnu: Using multiple sources of cqa-based information for answer selection and yes/no response inference. In *Proceedings of the 9th International Workshop on Semantic Evaluation, SemEval*, volume 15, 2015.