International Journal of Database Theory and Application
Vol.9, No.1 (2016), pp.35-44
http://dx.doi.org/10.14257/ijdta.2016.9.1.04

# Question Recommendation and Answer Extraction in Question Answering Community

Yang Xianfeng[1] and Liu Pengfei[2]

[1] *Yang Xianfeng, School of Information Engineering, Henan Institute of Science and Technology, Henan Xinxiang, China;*
[2] *HEBI Colleges of Vocation and Technology, Henan Hebi, China*
[1]*E-mail: 49377535@qq.com,* [2] *E-mail: 274772453@qq.com,*

***Abstract***

*Every day, there are a large number of new questions produce in question answering community, how to find the answer user for the question and sort candidate answers is the research content of this paper. First of all ,we use statistical language model to model for user interest, make full use of the abundant personalized information in question answering community to find out user interest distribution, and obtain the user list of question recommendation by introducing the query likelihood language model to calculate the degree of user interest to the new question. Secondly, we calculate the matching degree of question and candidate answers through fusing the feature of word form, word order, distance and semantic. The candidate answers of question will be sorted automatically, making it easier for users to choose the best answer. Experiments are performed on data sets extracted from the Baidu know, experimental results show that the method proposed in this paper has better performance.*

***Keywords****: question answering community, question recommendation, answer extraction, similarity calculation*

## 1. Introduction

In recent years, with the development of Web2.0 and social network services, Baidu know, Yahoo! answer and other question answering system based on community model has produced. Question answering community is a search system. It supports users to raise questions in natural language, while feedback to the user with more direct and concise answers. There is lots of questions and answers in question answering community. Therefore, with the increasing amount of data in the community, the efficiency of question recommendation and the accurate of answer extraction are key elements of question answering community research.

Many researches on question answering community has carried out, Artificial Intelligence Laboratory of University of Chicago developed FAQFinder, and they has done a lot of research work for English complex questions processing[1].City University of Hong Kong developed a practical CQA in 2007[2]. Research Institute of Microsoft in Asia retrieve question by identifying the theme and focus of question in CQA [3]. Emory University studied how to judge whether an answer can meet the needs of the questioner [4]. Meanwhile, some scholars predict user satisfaction for question answers in CQA [5]. After submitting question, they predict user satisfaction based on the answer time, the answer quality and other factors [6].

This paper studies question recommendation and answer extraction in question answering community. Through the construction of user interest model, question to be

ISSN: 2005-4270 IJDTA

solved will be recommended to those users who are interested in it, so that the question can be solved as soon as possible. In addition, according to the similarity calculation result of question and answer, candidate answers for question will be sorted automatically, so the questioner can more easily choose the best answer.

## 2. Question Recommendation Based on User Interest

### 2.1 Definition of Question Recommendation

Question recommendation is to recommend questions to be solved to those users who are interested in them, so that the question can be answered as soon as possible.

It is defined as follows: Given question set $Q = \{q_1, q_2, \cdots q_n\}$ and user set $U = \{u_1, u_2, \cdots, u_m\}$, for each user $u \in U$, question can be recommended to users, which satisfies the follow equation:

$$q_u = \arg\max_{q \in Q} Score_{q,u} \qquad (1)$$

Where $Score_{q,u}$ represents the degree of interest of the user $u$ to the question $q$.

In this paper, the algorithm flow of question recommendation is as shown in Figure1.

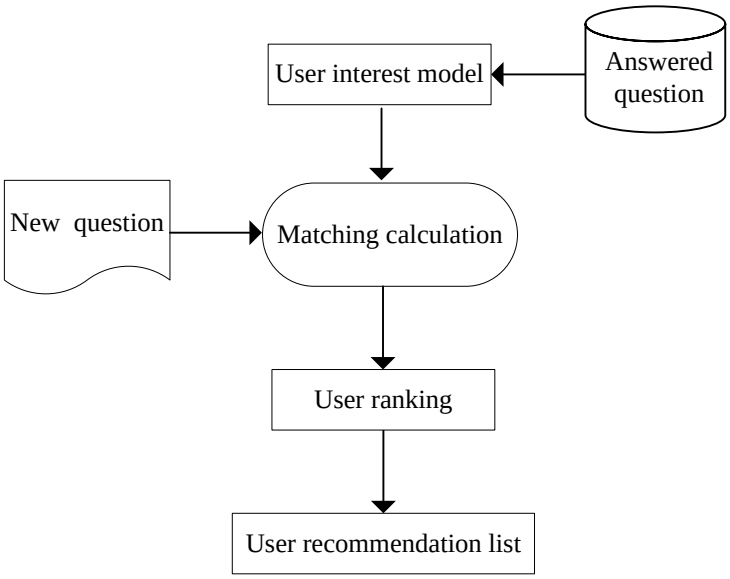


**Figure 1. The Algorithm Flow of Question Recommendation**

### 2.2 Construction of User Interest Model

In order to find the suitable respondents for new questions, it is need to construct user interest model. According to the matching degree of user interest and question, Questions can be recommended to suitable users. We can obtain user interest from answered questions of the user.

User interest is stable in a long time in question answering community, and users usually are only interested in questions with some specific topics. In a topic, the vocabulary used by the user typically has a greater similarity, while it has a greater difference in different topics. Therefore, we can obtain the degree of user interest for new question by calculating the similarity of answered questions and new question. If the similarity is high, it means that the user has a greater interest in this question, so, the user can be recommended to answer this question.

 

Therefore, we can use the language model to measure the degree of user interest by calculating the probability similarity of generating question from answered questions [7].

From the perspective of the language model, the degree of interest of the use $u$ to the question $q$ is defined as follows.

$$E(u,q) = P(q \mid \theta_{Q(u)}) \qquad (2)$$

Where $E(u,q)$ represents the degree of interest of the use $u$ to the question $q$, $P(q \mid \theta_{Q(u)})$ represents query likelihood in the language model $\theta_{Q(u)}$ of the user $u$ and answered questions $Q(u)$, it indicates the matching degree of the user $u$ and answered questions $Q(u)$.

The calculation method of user interest degree to question based on query likelihood model is as shown Figure2.

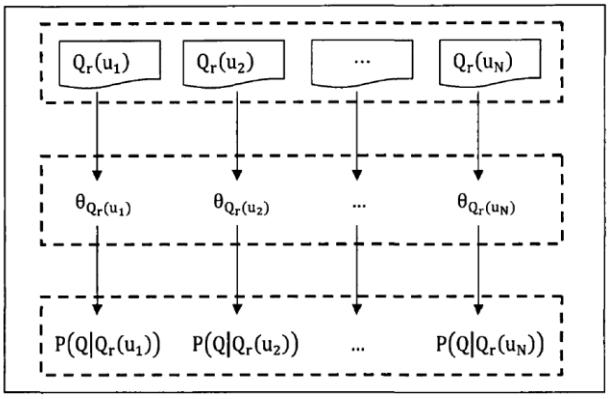


**Figure 2. Query Likelihood Model**

As can be seen from Figure 2, according to answered question set $Q_r(u_i)$ of all users, we first estimate the language model $\theta_{Q_r(u_i)}$, and then calculate the generation probability $P(q \mid \theta_{Q_r(u_i)})$ of question $q$ in the language model, finally, based on the value, rank for all users, and recommend suitable users to answer this question.

### 2.3 Estimation of Language Model

In order to calculate the query likelihood of the question *q*, firstly, we need to define language model of question set $Q(u)$, and then estimate language model based on question set. This paper uses polynomial distribution model to construct user language model. That is, the query is to be seen as a result sequence of randomized trials, each word *w* of the language corresponds to a random variable. The query likelihood $P(q \mid \theta_{Q(u)})$ is a polynomial distribution, for $q = \{w_1, w_2, \cdots, w_m\}$, it can be calculated as follows:

$$P(q \mid \theta_{Q(u)}) = \prod_{i=1}^{m} P(w_i \mid \theta_{Q(u)}) = \prod_{w \in q} P(w \mid \theta_{Q(u)})^{c(w,q)} \qquad (3)$$

Where $c(w,q)$ represents the number of times that the word appears in the question *q*.

In this paper, we use the maximum likelihood method to estimate the parameters of the model, that is $P(w \mid \theta_{Q(u)})$. The likelihood function is written in logarithmic form, which is expressed as follows.

$$\log P(Q(u) \mid \theta_{Q(u)}) = \log \prod_{w \in Q(u)} P(w \mid \theta_{Q(u)})^{c(w,Q(u))} = \sum_{w \in Q(u)} c(w,Q(u)) \log P(w \mid \theta_{Q(u)}) \qquad (4)$$

Where $c(w,Q(u))$ represents the number of times that the word appears in the question set $Q(u)$

By using the Lagrange multiplier method, we introduce a new variable $\lambda$ to combine the constraint conditions with the log likelihood function, and then get the Lagrange function.

$$L = \sum_{w \in Q(u)} c(w, Q(u)) \log P(w \mid \theta_{Q(u)}) + \lambda (1 - \sum_{w \in Q(u)} c(w, Q(u)) \log P(w \mid \theta_{Q(u)})) \quad (5)$$

L is to do partial derivative on $P(w \mid \theta_{Q(u)})$ and $\lambda$ respectively, and the partial derivative is set to 0. The maximum likelihood estimation for the language model can be expressed as follows.

$$P_{ml}(w \mid \theta_{Q(u)}) = \frac{c(w, Q(u))}{|Q(u)|} \quad (6)$$

Where $|Q(u)|$ represents the length of question set.

In order to eliminate the zero probability caused by data sparsity. This paper chooses Jehnek- Mercer smoothing method to perform linear interpolation for the language model of the question set. Finally, the estimation of language model is defined as follows.

$$P(w \mid \theta_{Q(u)}) = (1 - \lambda) P_{ml}(w \mid \theta_{Q(u)}) + \lambda P(w \mid C) \quad (7)$$

Where $\lambda$ represents the smoothing parameter, $\lambda \in [0,1]$, $p(w \mid C)$ represents the language model of the entire data set.

The query likelihood which is also the degree of interest of user to new question is expressed as follows.

$$\begin{aligned} Score_{q,u} &\propto P(q \mid \theta_{Q(u)}) = \prod_{w \in q} P(w \mid \theta_{Q(u)})^{c(w,q)} = \prod_{w \in q} P_{\lambda}(w \mid \theta_{Q(u)})^{c(w,q)} \\ &= = \prod_{w \in q} [(1 - \lambda) \frac{c(w, Q(u))}{|Q(u)|} + \lambda \frac{c(w, C)}{|C|}]^{c(w,q)} \end{aligned} \quad (8)$$

Thus, the question list for the user $u$ can be obtained by sorting $Score_{q,u}$. The first n questions can be selected to recommend to the user $u$.

## 3 Answer Extraction Based on Feature Fusion

### 3.1. Definition of Answer Extraction

Answer extraction can automatically choose the best answer from the candidate answers of question. It helps users determine the best answer for a question, and also makes it possible for predicting the best answer automatically in question answering community. Answer extraction is defined as follows.

Given the question $q$ and its candidate answer set $A_q = \{a_1, a_2, \cdots a_n\}$, the best answer $a_i \in A_q$ selected from the answer set is expressed as follows.

$$a_i = \arg\max_{a \in A_q} Score_{q,a} \quad (9)$$

Where $Score_{q,a}$ represents the matching degree of the answer $a$ and the question $q$.

In this paper, the algorithm flow of answer extraction is shown in Figure 3.

 

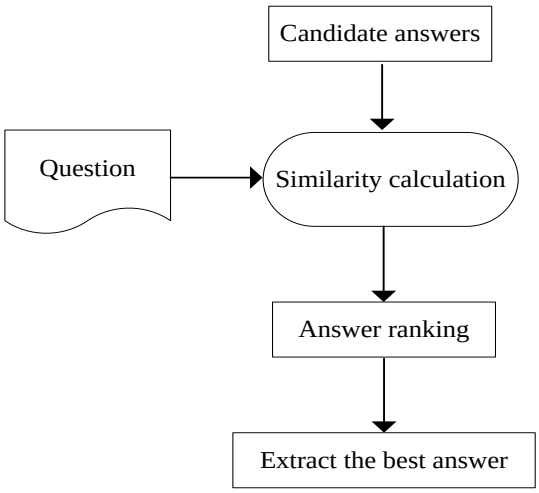


**Figure 3. The Algorithm Flow of Answer Extraction**

### 3.2. Similarity Calculation

Similarity of answer and question is actually calculate the similarity between sentences, which influenced by many factors. This paper mainly considers four aspects: word form, word order, distance and semantic. The definition and calculation method is given below.

Definition 1 word form similarity *WordSim (q, a)*

Word form similarity is the similarity degree of question and answer in shape, which is measured by the number of common words containing in two sentences. It can be calculated as follows:

$$WordSim(q,a) = \frac{Same(q,a)}{Word(q) + Word(a) - Same(q,a)} \qquad (10)$$

Where *WordSim (q, a)* represents the number of the same keywords in the answer *a* and the question *q*. If the same keyword appears more than once, it can be counted only once. $Word(q)$ represents the number of keywords in question q and $Word(a)$ represents the number of keywords in answer a.

Definition 2 word order similarity *OrdSim(q,a)*

This method is to mark the similarity of sentences from the order of keywords. It reflects the similarity degree of the same words or synonyms of two sentences in the position, which can be measured by the number of the adjacent sequence inverse of the same words or synonyms. The calculation method is as follows.

$$OrdSim(q,a) = 1 - \frac{\mathrm{Re}v(q,a)}{Max\,\mathrm{Re}v(q,a)} \qquad (11)$$

Where $Max\,\mathrm{Re}v(q,a)$ represents the maximum reverse of natural sequence that the question and answer has the same number of keyword, *Rev(q,a)* represents the reverse of natural sequence that is constituted by keywords of the question *q* in the position of the answer *a*.

Definition 3 distance similarity *DisSim (q, a)*

Distance similarity is to measure the similarity by calculating the distance of the same keywords in the question and answer. It uses *DisSim (q, a)* to express the distance similarity between question *q* and answer *a*, which is calculated as follows:

$$DisSim(q,a) = 1 - abs\left|\frac{SameDis(q) - SameDis(a)}{Dis(q) + Dis(a)}\right| \qquad (12)$$

Where $SameDis(q)$ represents the distance of the same keywords of question $q$ and answer $a$ in the question $q$, if the same keyword repeat several times, we use the maximum distance. *Dis (q)* represents the distance of keywords in the leftmost and the rightmost whose are no repeated keywords in question $q$, if the keyword appears several times, we use the minimum distance.

Definition 4 semantic similarity

Semantic similarity calculation is based on the calculation of word semantic. The reference [8] introduced the similarity calculation based on HowNet. We use this semantic calculation method in this paper.

Given the question $q$ and the answer $a$, $q$ contains these keywords $w_{11}, w_{12}, \cdots w_{1n}$ , $a$ contains these keywords $w_{21}, w_{22}, \cdots w_{2m}$. So the similarity between the keyword $w_{1i}(1 \le i \le n)$ and $w_{2j}(1 \le j \le m)$ can be expressed as $Sim(w_{1i}, w_{2j})$ . Semantic similarity between the question $q$ and the answer $a$ can be calculated as follows.

$$SemSim(q,a) = \frac{1}{2}(\frac{1}{n}\sum_{i=1}^{n}\max\{Sim(w_{1i}, w_{2j}) \mid 1 \le j \le m\} + \frac{1}{m}\sum_{i=1}^{m}\max\{Sim(w_{1i}, w_{2j}) \mid 1 \le i \le n\}) \quad (13)$$

Definition 6 similarity calculation based on feature fusion

In summary, the similarity of question and answer is determined by the above factors. Therefore, this paper fuses the above four kinds of features, the similarity of question and answer is calculated as follows:

$$Score_{q,u} \infty Sim(q,a) = \lambda_1 WordSim(q,a) + \lambda_2 OrdSim(q,a) + \lambda_3 DisSim(q,a) + \lambda_4 SemSim(q,a) \quad (14)$$

Where $\lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 = 1$ and $\lambda_4 \ge 0.5 \ge \lambda_1 \ge \lambda_2 \ge \lambda_3$

## 4 Experimental Results and Analysis

### 4.1. Experimental Data

In this paper, the experimental data selects from the question answering community-Baidu know. According to the number of questions in each category, we sorted categories in descending order and select the front 6 categories. The number of questions in each category is more than 1000, and the number of answers for each question is more than 5. The best answer is marked by 6 members of group, where the best answer of each question is marked by three people. The experimental data is as shown in Table 1. In addition, most users only answered very few questions, so, it is unable to learn their interest model. Users recommended by system should be more active respondents and they have a certain historical answer record. Therefore, this paper selects users who answer questions more than 15 times, and constructs user data set according to the user's answer information.

**Table 1. Information of Experimental Data**

| The number of questions | The number of answers | The number of respondents | The number of categories | The average number of answers for each question |
|---|---|---|---|---|
| 12546 | 109792 | 14873 | 6 | 8.75 |

### 4.2. Evaluation Tools

Text information processing results has many evaluation standards. The most common is the precision rate and recall rate [9]. Because of the particularity of the question

 

answering community, the recall rate can be ignored in the result analysis [10]. The precision rate does not consider the order of recommendation result. Therefore, it generally uses the average precision rate instead. That is:

$$AvgPrec = \frac{1}{N}\sum_{i=1}^{N}\frac{i}{r_i} \qquad (15)$$

Where $N$ represents the number of recommendation result, $r_i$ represents the ranking position of the *i-th* related result, if not related, $r_i$ is infinite.

Since this experiment have more than one test data, therefore, this paper use mean average precision (MAP) as the evaluation index of the recommendation algorithm.

In order to measure the average number of recommendation users, the question can be recommended to the relevant users. Taking into account the order of the relevant users, this paper adopts the MRR evaluation method for further evaluation. The calculation formula is as follows:

$$MRR = \frac{1}{|Q|}\sum_{q\in Q}\frac{1}{r_q} \qquad (16)$$

Where $|Q|$ represents the number of questions， $r_q$ represents the ranking position of the first returned result of the question $q$.

### 4.3. Result Analysis

In this experiment, the first step is to do Chinese word segmentation on user data and model user interest. For the construction of user interest model based on language model, we used formula (8) to calculate the user score. Experimental results show that the smoothing parameter has the best effect when $\lambda = 0.2$ . Therefore, in subsequent experiments, this paper fixed $\lambda = 0.2$ to model user interest, and accordingly recommended. The experimental result is as shown in Figure 4.

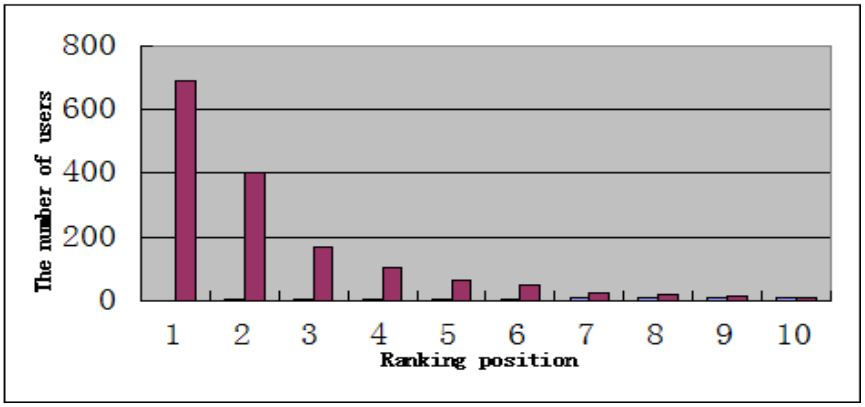


**Figure 4. Recommendation Ranking Position of the Best Answer User**

As shown in Figure 4, based on the ranking statistical results of the best answer user in question candidate users, we can see that most of the best answer users is in the front of ranking list. The experimental results show that the proposed method of question recommendation based on user interest has high accuracy.

In order to evaluate the performance of answer extraction method proposed in this paper, we use four different retrieval models to perform comparative experiments: (1) wordsim: similarity calculation on the basis of word form. (2) wordsim+ordsim: the introduction of word order information on the basis of the first model. (3) wordsim+ordsim+dissim: the introduction of distance information

on the basis of the second model. (4) ordsim+ordsim+dissim+semsim: the introduction of semantic information on the basis of the third model, namely the proposed method. The value of experimental parameters is set $\lambda_1 = 0.2,\ \lambda_2 = 0.1,\ \lambda_3 = 0.1,\ \lambda_4 = 0.6$ . The experimental result is as shown in Table 2.

**Table 2. Experimental Results of Different Experimental Methods**

| Experimental method | MAP | MRR |
|---|---|---|
| wordsim | 0.624 | 0.815 |
| wordsim+ordsim | 0.641 | 0.831 |
| wordsim+ordsim+dissim | 0.667 | 0.848 |
| ordsim+ordsim+dissim+semsim | 0.705 | 0.867 |

As it can be seen from the Table 2, we can get a higher average precision when only using the method of wordsim. This is because the extracted data sets from Baidu know are the most popular categories, answers associated with each question will be more. With the factors of word form, word order, distance and semantic added to experiment sequentially, MAP and MRR are also rising. It indicates that utilization of word form, word order, distance and semantic can more fully excavate information to be expressed between question and answer. It is benefit to calculate the similarity between sentences, and then extract the best answer to the question. This also shows the proposed method is feasible and effective.

## 5. Conclusion

This paper explores the question recommendation and answer extraction in question answering community. Through modeling user interest, we find out those users who are interested in the question, and recommend the question to them. When modeling, the characteristic of linguistic knowledge is used comprehensively, and the information of sentence structure is fully excavated. This paper also analyzes the candidate answers, calculates the similarity between the question and the answer by fusing the multiple features of syntactic and semantic, and then gets the recommended answer list, so the questioner can more easily choose the best answer. Results of comparative experiments show that the proposed method can effectively recommend questions and extract answers. Because of the influence of the classification precision, considering the specific field, the next research content is to design question recommendation and answer extraction algorithm for the field.

## References

[1] N. Tomuro, "Interrogative Reformulation Patterns and Acquisition of Question Paraphrases", Proceeding of the Second International Workshop on Paraphrasing, July **(2003)**, pp. 33-40.
[2] S. Wanpeng, F. Min and G. Naijie, "Question Similarity Calculation for FAQ Answering", Third International Conference on Semantics, Knowledge and Grid, **(2007)**, pp. 298-231.
[3] D. Huizhong, C. Yunbo and L. Chinyew, "Searching Questions by Identifying Question Topic and Question Focus", Proceeding of ACL, **(2008)**, pp. 156-164.
[4] L. Yandong, B. Jiang and E. Agichtein, "Predicting Information Seeker Satisfaction in Community Question Answering", Proceeding of SIGIR, New York, USA, **(2008)**, pp. 483-490.
[5] A. Agarwal, H. Raghavan and K. Subbian, "Learning to Rank for Robust Question Answering", Proceedings of the 21st ACM international conference on Information and knowledge management, New York, USA, **(2012)**, pp. 833–842.
[6] B. Li, I. King and M. R. Lyu, "Question Routing in Community Question Answering: Putting Category in Its Place", Proceedings of the 20th ACM Conference on Information and Knowledge Management, Glasgow, Scotland, October **(2011)**, pp. 2041-2044.
[7] J. M. Ponte and W. B.Croft, "A language modeling approach to information retrieval", Proceedings of the 21nd International ACM SIGIR Conferenee on Research and Development in Information Retrieval, Melbourne, Australia, **(1998)**, pp. 275-281.

 

[8] L. Qun and L. Sujian, "Lexical semantic similarity calculation based on HowNet", Proceedings of the Third Chinese Lexical Semantics Workshop, **(2002)**, pp. 8-15.
[9] B. M. John, A. Y. Chua and D. H. Goh, "What makes a high-quality user-generated answer", IEEE Internet Computing, vol. 1, no. 15, **(2011)**.
[10] Z. Zhu, D. Bernhard and I. Gurevyeh, "A multi-dimensional model for assessing the quality of answers in Social q & a sites", Technical, **(2010)**.

 