Information Sciences 596 (2022) 395–413

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

High-quality domain expert finding method in CQA based on multi-granularity semantic analysis and interest drift

Yue Liu [a,b,c,*], Weize Tang [a], Zitu Liu [a], Lin Ding [a], Aihua Tang [a]

[a] School of Computer Engineering and Science, Shanghai University, Shanghai 200444, China
[b] Shanghai Institute for Advanced Communication and Data Science, Shanghai University, Shanghai 200444, China
[c] Shanghai Engineering Research Center of Intelligent Computing System, Shanghai 200444, China

ARTICLE INFO

Article history:
Received 18 December 2020
Received in revised form 17 February 2022
Accepted 20 February 2022
Available online 2 March 2022

Keywords:
Community question answering
Domain expert finding
Semantic analysis
Interest drift
User quality

ABSTRACT

Expert finding is an important research field in community question answering (CQA). Traditional expert finding methods mainly exploit topic analysis and authority calculation methods to identify high-quality experts in certain fields. To avoid recommending questions to those experts who do not display the willingness or ability to provide high-quality answers, user interest drift and user quality should be considered. This study proposes a novel method named high-quality domain expert finding in CQA based on multi-granularity semantic analysis and interest drift (HQExpert). Firstly, HQExpert considers different semantic granularities by employing two models, a coarse-grained topic model LC-LDA and a fine-grained model (BERT), to capture the domain information of questions and users more accurately. Secondly, to address the diverse interests of the users, a user interest drift model in HQExpert is developed to dynamically represent the changes in the interests of the users at different periods. In addition, a user quality model is developed to further optimize the professional level of the user, finding experts who can provide high-quality answers and are interested in the current question. Finally, extensive experiments on two datasets from different domains demonstrate that the proposed HQExpert model can significantly improve the accuracy of finding high-quality experts.

1. Introduction

Community Question Answering (CQA) websites, such as Stack Overflow and Baidu Knows, have gradually become very popular interactive platforms for sharing knowledge in recent years. The major task of CQA is expert finding [1], which is to find the appropriate and willing users who have the relevant expertise to provide high-quality answers to the given questions.

In the past decade, many researchers focused on the field of expert finding in CQA and proposed many different expert finding methods [2–9]. These existing methods can be classified into topic-based methods, authority-based methods, or a combination of the two types. The topic-based methods [8,10–12] represent both the domain of expertise of the user and the question content by latent topic modelling techniques; the similarity between them is then calculated to find experts in similar domain. The authority-based methods [13,14] exploit user-to-user interactive networks based on the asking-answering relation of users to calculate user authority by exploiting link analysis (e.g., PageRank [15], HITS algorithm

* Corresponding author at: School of Computer Engineering and Science, Shanghai University, Shanghai 200444, China.
E-mail address: yueliu@shu.edu.cn (Y. Liu).

https://doi.org/10.1016/j.ins.2022.02.039

[16]) while ignore the domain similarity between questions and answerers. Recently, some studies have combined topic-based methods and authority-based methods [17–19] to exploit high-level experts in similar domains. Though these methods have achieved good results, there are some new features (e.g., deep context semantics, expert interest drift, and the quality of experts) that need to be considered. Using these features effectively is new challenge that can improve the performance of expert finding in CQA. Thus, the motivation of this study is to construct an effective expert finding method, which can retrieve the expert with both willingness and high quality to enhance the experience of the user in CQA.

Topic-based expert finding methods can find the relevant expertise of users by analysing the semantics of their answer records from the perspective of the topic whilst ignoring the fine-grained semantic of the current question and the answer records of the users. Recently, novel fine-grained semantic representation models, such as Word2vec [20] and FastText [21] have been proposed. They can translate the context of the current question and the answer records of the users into a fine-grained semantic representation, effectively representing the semantics of the text on a finer granularity than topic-based models. Our latest question retrieval method, named WELQLC-QR [22], embedded topic information into the word2vec model [20] with word importance to achieve more accurate semantic representation, which also proved that multi-granularity was an effectively method for comprehensive and accurate analysis of question content and the domain of the expert. However, the word2vec model is considered to be context independent. The static word representation by word2vec is difficult to effectively identify polysemy and accurately represent the semantics of a word in its context. Thus, capturing the semantics of polysemy in different context and accurately encoding a question or a user for generating multi-granularity domain representations are major challenges.

The willingness of users should be taken seriously in expert finding methods [23]. The willingness of experts will change with their interests, which are often dynamic in CQA. From August 2008 to September 2014, there were only 7% of users with one category of interest and more than half with three or more categories of interest from the Programmer sub-forum dataset in Stack Overflow [24]. This indicates that the diversity of user interests in CQA is universal. Besides, users might gradually become interested in other fields in the process of accumulating professional knowledge. Taking the user with ID “27617” in Stack Overflow as an example, Fig. 1 shows that the interests of this user have gradually changed from “Python” to “Java” over the past few years. However, how to model user interest drift over time and integrate them into the multi-granularity representation to represent the current interest of the users are research priorities in this paper.

Furthermore, low-quality experts can hardly provide high-quality answers to the questions of other users. Therefore, the quality of the experts should also be considered in the process of expert finding. The honour system in CQA provides the evaluation information of the users, which is determined by other users and can more intuitively reflect the professional level of the users. Some researchers preliminarily took answer quality into consideration during expert quality analysis in CQA [2,23]. Based on this, we further take advantage of more evaluation information (e.g., the number of views, likes, and the honour score) provided by the honour system. One key challenge is that these evaluation features often have large differences in data format and dimension and cannot be directly used to construct a user quality model. Therefore, constructing a user quality evaluation model to scientifically and comprehensively characterise these features and optimizing the existing expert professional level calculation method for improving effects are big challenges.

To address the above-mentioned challenges, we propose a high-quality domain expert finding method in CQA based on multi-granularity semantic analysis and interest drift (HQExpert). First, to dynamically encode a question in different contexts, HQExpert introduces the stronger pre-trained language model (BERT model [25]), in order to achieve fine-grained domain representations, and combines it with coarse-grained topic representations obtained by labelled clustering-latent Dirichlet allocation LC-LDA [26] to generate multi-granularity and contextual domain representations. Therefore, information on the domain of the expert and question content can be accurately expressed on two granularities to find the closest expert in the target domain. Furthermore, in order to capture the dynamic changes of user interests over time, a user interest

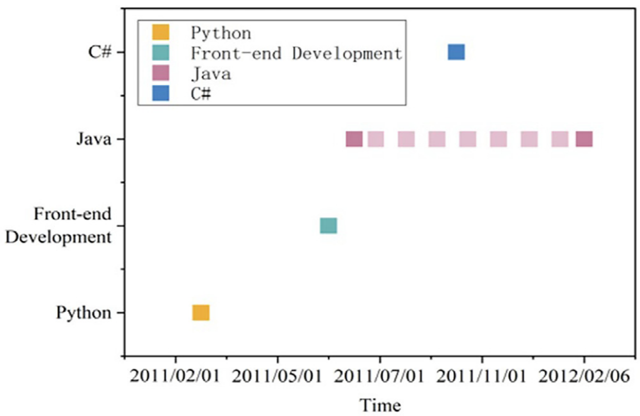


Fig. 1. Changes of question type answered or asked by user “27617” against time. After February 1, 2011, user “27617” has not generated any asking-answering records regarding Python. One year later, the user just generated asking-answering records regarding Java.

drift model is proposed to dynamically portray the time-sensitive interest of the user. The main function of the interest drift weight is to modify the multi-granularity semantic representation of the domain of the expert and further mine the experts who are willing to answer the given questions. Specifically, to solve the quality of experts by domain matching not being high enough problem, a multi-angle and multi-dimension user quality model is constructed to evaluate the quality of the experts and then optimize the TSWPR algorithm [26], which is an authority-based method for identifying high-quality experts with high professional level. Finally, we carry out comparative experiments along with benchmark models and counterpart methods on two real-world CQA datasets from Stack Exchange and Stack Overflow to demonstrate the effectiveness of our proposed method.

The remainder of this paper is organized as follows. Section 2 surveys the related works on expert finding and user quality analysis. Section 3 details the main ideas and key techniques of the proposed method. Section 4 describes a variety of experiments and results to verify the superiority of the proposed framework (HQExpert). Finally, conclusion and future work are presented in Section 5.

2. Related work

2.1. Expert finding

Traditional expert finding methods can be divided into two categories: topic-based methods and authority-based methods.

Topic-based methods employ latent topic modelling techniques to find domain-related experts. Daud et al. [27] simultaneously modelled conferences influence and time information and proposed a semantic based temporal-expert-topic approach and temporal information based expert search for temporal expert finding. Guo et al. [8] and Riahi et al. [1] exploited a latent Dirichlet allocation (LDA) topic model in the process of semantic matching, which can avoid the limitation of semantic matching brought by "parallel hypothesis" and improve the accuracy of expert finding to a certain extent. Momtazi et al. [28] proposed a topic modelling approach by using LDA to induce probabilistic topics and presented the superiority of the proposed topic-based approach to improved document-based expert finding systems. However, it is not accurate enough for the above methods to treat the users whose interest domains semantically close to a given question as "experts", in which experts are in different professional levels.

Authority-based methods are based on the link analysis of the historical activities of the experts to discover high-authority experts. Jurczyk et al. [13] introduced an HITS algorithm for expert finding. They constructed a user network based on the asking-answering relations among users, and used the HITS algorithm to evaluate hub and authority values to find a group of experts with a high professional level. Zhang et al. [29] further introduced PageRank [15] in collaboration with HITS. They analysed the user network in CQA in more detail and proposed a social network analysing method to evaluate the professional level of experts. By extending the PageRank algorithm, Zhou et al. [30] proposed a topic-sensitive probabilistic model and more effectively found experts in the community by incorporating link and user analysis into a unified framework. To learn the complex information contained in a user network, Li et al. [31] learned the representations of a question, question raiser, and answerer by using a heterogeneous information network embedding algorithm. However, the above-mentioned methods based on link analysis consider only the professional level of users and ignore the field of experts.

To find high-quality experts in target domain, some researchers proposed various expert finding methods that combined topic-oriented approaches with authority-based approaches. Zhu et al. [19] measured the topic relevance of questions and ranked the user professional level in a user-to-user link graph. Kao et al. [32] proposed a hybrid expert finding model. This model firstly used question category information to represent the domain information of the user, used PageRank to calculate the professional level of the user, extracted the honour values, and finally ranked the experts based on these three metrics to identify effective experts. Geerthik et al. [33] proposed a domain expert ranking (DER) model to identify experts based on commonly available parameters in CQA. These parameters included domain cut-off, answer view rank, common domain relationship, followed domain, previous answers, week visit, computing the DER, and the levels of experts with DER. Liu et al. [26] employed LC-LDA (labelled clustering-LDA) to extract the topic categories of questions and answerers, further considered the difficulty of those questions, and proposed a label clustering topic-sensitive weighted PageRank algorithm (TSWPR, an improved topic-sensitive PageRank algorithm) to find high-quality users users. Nobari et al. [7] defined a concept (i.e., Voteshare) and used it in the expert finding task. They found that utilizing statistical and word embedding translation approaches and considering the quality-aware scoring simultaneously could significantly improve the quality of expert finding in StackOverflow.

Significantly, this study conducts a comprehensive semantic analysis of texts from experts and questions defined from multiple semantic granularities to find domain-related experts accurately. Furthermore, this paper also proposes the interest drift model and the quality model of users to identify appropriate experts.

2.2. User quality analysis

Some researchers preliminarily extracted the professional level of the user to characterize the user quality based on graph algorithms [34] and enhanced the effectiveness of the expert finding methods. For example, Zhang et al. [29] used the network influence of the users as the professional level of the users in CQA by simply exploiting the PageRank algorithm. Li et al. [2] further proposed and considered the historical answer records of user, including the number of answers, asking, comments to evaluate the quality of the user. However, for the task of expert finding, this evaluation method is likely to mistake the "indiscriminate" users who have answered many questions with low quality as experts. At the same time, the feedback from others is also significant in finding experts. The feedback refers to user evaluation index (including the numbers of likes) provided by the honor system in CQA, which can intuitively evaluate the user quality. Therefore, to measure the quality of users in the area of expert finding more scientifically, Wang et al. [35] further considered the number of likes of answer to find high-quality experts who could provide high-quality answers. Additionally, Liu et al. [26] evaluated user quality from the topic of the text and the number of likes. In this way, "indiscriminate" users with similar topic could be identified. Note that the algorithm is mainly proposed to predict the response time of high-quality answers, rather than the task of expert finding. Apart from the number of likes, the number of views and the reputation score that can directly reflect the quality of users in CQA should be also considered.

Different from traditional user quality analysis from single evaluation index, to effectively quantify the quality of users, a multi-angle user quality model is proposed, comprehensively considering the number of views, the number of likes, and the score of reputation of each user in a unified way. Furthermore, we use the comprehensive user quality and user authority to identify high-quality and professional experts.

3. Methods

In order to effectively find domain experts, a high-quality domain expert finding method in CQA based on multi-granularity semantic analysis and interest drift (HQExpert) is proposed, and this method consists of the following two modules (Fig. 2).

- Expert domain representation based on multi-granularity semantic analysis and interest drift. Firstly, HQExpert uses LC-LDA and BERT to obtain the semantic representation of the expert and question at both the coarse and fine-grained levels, and then accurately represents the domain information of the expert and the context content of the question. Secondly, it extracts the potential interest distribution of the users according to their activity records and obtains the weight of their interest drift. Subsequently, the extracted potential interest distribution is used to modify the domain representation of experts, which can effectively characterize the current interest domain of the experts. Finally, the modified expert domain representation is matched with the semantic representation of the question to find the relevant domain experts.
- Expert ranking strategy based on quality optimizing TSWPR (RKExpert). Firstly, it builds a user quality model based on user honour values (e.g., the number of views, likes and answers) in CQA, and a user interactive network based on interactive data. Afterwards, the weight of the interactive network is updated by the user quality model and other information (e.g., the score of the question and answer, the acceptance of the question, the topic similarity between question and answer). Based on the model and the network, a quality optimizing TSWPR algorithm is used to obtain the user professional level. Finally, the user professional level is exploited to update the results of qualified domain expert retrieval.

3.1. Domain representation based on multi-granularity semantic analysis and interest drift

This section mainly performs expert finding from the perspective of domain matching, which exploits the content of given questions to find experts in related fields.

3.1.1. Domain representation based on multi-granularity semantic analysis

The existing domain expert finding methods transform an expert finding problem into an information retrieval problem. By matching the content of a question with the text that can represent the domain of experts, the experts in related fields can be found. In order to represent the domain information of experts comprehensively and accurately, this study introduces fine-grained semantic analysis of experts in addition to the traditional coarse-grained expert domain representation method based on the topic model.

First of all, the following assumptions are provided on the interpretation of the user domain.

Hypothesis 1. Users generally tend to answer questions in their biased fields. Therefore, the answers given by users in CQA can directly reflect the relevant fields of the users.

Topic model is a kind of coarse-grained semantic matching algorithm, which can represent text content in CQA from the perspective of coarse-grained semantic representation. In this study, LC-LDA is used to extract question semantic representation. LC-LDA can be roughly divided into two steps: clustering labels (tags attached to questions) to alleviate the impact of low frequency tags on results and then exploiting L-LDA to learn topic distribution of questions and answers. This paper will introduce the process of label clustering. A label co-occurrence probability matrix C_{n*n} [26] is constructed according to collection of labels (tags attached to questions) $T^{(n)}$ in CQA, where n represents the number of labels, and each element c_{ij} of the matrix represents the probability that the label t_i and t_j appear at the same time.

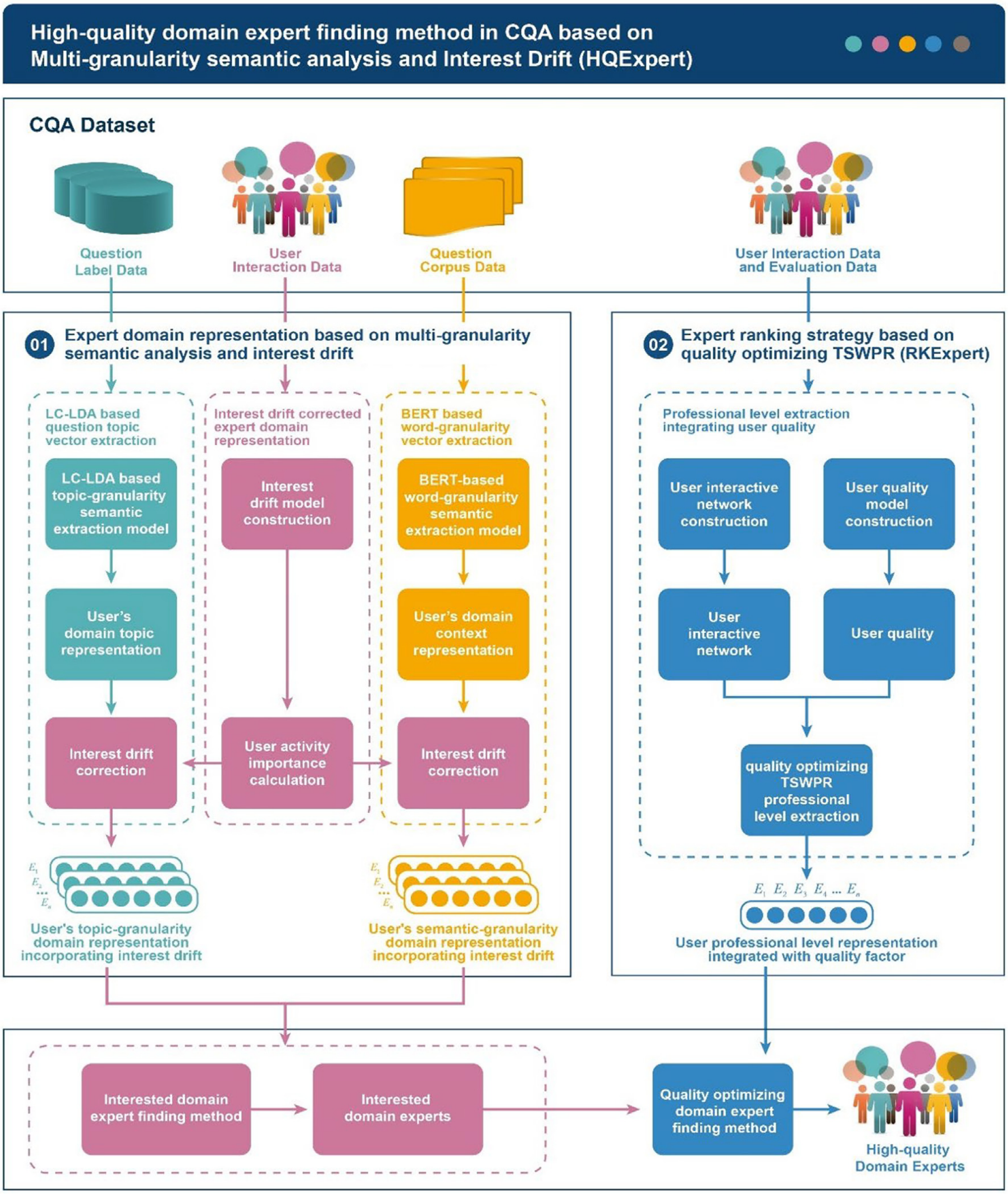


Fig. 2. Framework of HQExpert consists of expert domain representation and RKExpert. Expert domain representation is used to obtain the current domain representation of users and find appropriate domain experts. RKExpert is used to obtain user professional level to update the results of the qualified domain expert retrieval.

$$c_{ij} = \frac{u_{ij}}{\sum_{k=1}^{n} u_{kj}}, \tag{3-1}$$

where u_{ij} represents the co-occurrence times of t_i and t_j, and $\sum_{k=1}^{n} u_{kj}$ represents the occurrence times of t_j.

The "expansion" and "inflation" operations are performed alternately when constructing the matrix C_{n*n} until it converges (i.e., the matrix no longer changes with "expansion" and "inflation"). "Expansion" refers to the product of the matrix, as

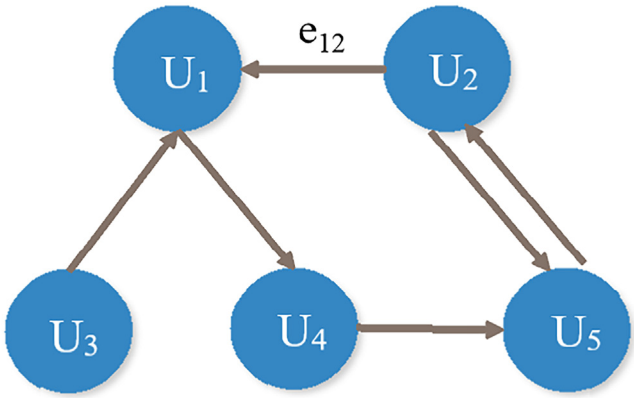


Fig. 3. User interactive network. The blue nodes represent users while the grey edges represent question-answer relationships between users. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

shown in Eq. (3-2). In the process of random walk, "expansion" represents the probability that the walker stays at each node at the next moment, which is the process of Markov chain iteration [36,37].

$$C_{exp=}Expand(C) = C \cdot C \tag{3-2}$$

Correspondingly, "inflation" refers to actively stopping the process of the Markov chain iteration after expanding each element c_{ij} of the matrix to r power. Eq. (3-3) is used to normalize each column of matrix C_{n*n} after "expansion" operation.

$$c_{ij}^{(inf)} = \frac{c_{ij}^r}{\sum_{k=1}^{n} c_{kj}^r}, \tag{3-3}$$

where $r > 1$ represents the expansion coefficient.

After performing "expansion" and "inflation" operations multiple times on the matrix, C_{n*n} begins to converge. In the resulting matrix C, the non-zero value c_{ij} in the j-th column indicates that label t_i is the cluster center of label t_j, that is to say, t_i and t_j belong to the same category. After clustering the labels through MCL, new label category vectors can be obtained.

When the probability transition matrix C_{n*n} converges, the matrix C_{n*n} can be interpreted that if some columns are not equal to zero in the same row, the labels corresponding to these columns belong to the same cluster. Consequently, K label clusters are obtained. Then, we use a list of binary cluster presence/absence indicators $\Lambda^{(d)} = (l_1, \cdots, l_K)$ to indicate the affiliation of question d. If $l_i = 1$, question d belongs to the i-th label cluster. Otherwise, it does not. Afterwards, the above cluster vector $\Lambda^{(d)}$ is employed as the training annotations for L-LDA model [11,37] to obtain the topic distribution $Z^{(q)}$. Therefore, given a query question set Q and an answer set A provided by N users of CQA, the topic distribution $Z^{(q^m)} = \left(z_1^{(q^m)}, z_2^{(q^m)}, \cdots, z_K^{(q^m)}\right)$ of each question q^m and the topic distribution $Z^{(a^n)} = \left(z_1^{(a^n)}, z_2^{(a^n)}, \cdots, z_K^{(a^n)}\right)$ of each answer a^n can be calculated by the LC-LDA model.

To express the semantic of a question from a more fine-grained perspective, this study also uses BERT model to extract the semantic representation of expert domain. Compared with the traditional fine-grained semantic extraction models such as Word2vec and FastText, the most notable characteristics of BERT is that it can recognize polysemous words (i.e., the different meanings of a word in different contexts). In expert finding, the domain representation of an expert is usually a long text, which is prone to ambiguity. The introduction of BERT can more accurately extract the semantics of the text, thereby promoting the accuracy of expert finding.

BERT is a model based on encoder-decoder architecture [38], consisting of encoding and decoding. Different from general encoder-decoder models, the encoder and decoder of BERT are not composed of the traditional deep learning model, i.e. recurrent neural network (RNN) [39], but a bilateral transformer encoder.

In this model, $E_1, E_2, \cdots, E_N$ represent input texts, which are encoded and decoded by Transformer, and a text vector with semantic information is obtained.

Using the transformer as an encoder can accelerate the training process in parallel. At the same time, the self-attention mechanism [38] is adopted in the process of semantic extraction considering context information, allowing for the same word with different semantic representation to be extracted in different contexts.

Given a question corpus q of CQA, q = $(w_1, w_2 w_n)^T$after word segmentation, where n represents the number of words in the question, and w_i represents the i-th word in the question. Then, each word is represented by a k-dimensional one-hot encoding. Thus, the question is transformed into $A = \left(a^1, a^2, \ldots, a^n\right)^T$. A is a $n \times k$ matrix, and each row of A is the one-hot encoding vector representation of a word in the question corpus.

Next, the three key matrixes Q, K, and V in the self-attention mechanism need to be calculated by Eq. (3-4) [38].

$$\begin{aligned} Q &= AW^{Q} \\ K &= AW^{K}, \\ V &= AW^{V} \end{aligned} \tag{3-4}$$

where W^{Q}, W^{K}, and W^{V} are three weight matrices for calculating Q, K, and V, respectively, the size of both W^{Q} and W^{K} is $k \times d_k$, and the size of W^{V} is $k \times d_v$. These three matrices are obtained during the training process, and the sizes of matrixes Q, K and V are $n \times d_k$, $n \times d_k$ and $n \times d_v$ respectively. Each row of Q, K and V represents the vector of each word in the input question corpus.

With the three matrices, the size of attention can be calculated by Eqs. (3-5) and (3-6).

$$Attention\ (Q, K, V) = Softmax\left(\frac{Q \cdot K^{T}}{\sqrt{d_k}}\right) \cdot V, \tag{3-5}$$

$$Softmax\ ((z_1, z_2, \cdots, z_N)) = \frac{1}{\sum_{i}^{N} e^{z_i}}\ (e^{z_1}, e^{z_2}, \cdots, e^{z_N}), \tag{3-6}$$

The above Softmax function is used for normalization, scaling the row elements in a matrix to [0,1]. It can be seen that the attention is essentially a calculated weight [38]. In the process of extracting a semantic vector, attention calculates the correlation between each word and the other words in the text. Then it integrates the correlation into the process of semantic extraction, so that the final semantic vector of a word can contain context information. For a word, its semantic vectors extracted in different contexts are different, thereby overcoming the influence of polysemy.

Based on the attention mechanism, the transformer further proposes a multi-head self-attention mechanism [38], shown in Eqs. (3-7) and (3-8).

$$MultiHead(Q, K, V) = Concat(head_1, head_2, \cdots, head_k)W^{O}, \tag{3-7}$$

$$head_i = Attention(QW_i^{Q}, KW_i^{K}, VW_i^{V}), \tag{3-8}$$

where W_i^{Q}, W_i^{K}, and W_i^{V} represent the three weight matrices of the *i-th* head. Through horizontal splicing, self-attention can focus on abundant semantic information.

Specifically, through the BERT model, the fine-grained semantic representation $v^{(q^m)} = \left(x_1^{(q^m)}, x_2^{(q^m)}, \cdots, x_n^{(q^m)}\right)$ of each question q^m and $v^{(a^n)} = \left(x_1^{(a^n)}, x_2^{(a^n)}, \cdots, x_n^{(a^n)}\right)$ of each answer a^n can be obtained.

3.1.2. Time-sensitive domain representation incorporating interest drift

The interests of experts in CQA are not always the same; hence, a user interest drift model is built to represent the change in the interests of the users at different periods. The construction of the interest drift model is essential to measure the influence of the current answering activity on the potential interest of the user within a period. Here, the answering activity influence is introduced to quantify the influence of the historical answering activities of the users on their current potential interest.

Definition 1. Answering Activities Influence: It represents the influence of user $\mathrm{u_i}$'s response behaviour $\mathrm{e_{t_j}^{u_i}}$ at time $\mathrm{t_j}$ on the current interest of the user. It is denoted as $\mathrm{R}\left(\mathrm{e_{t_j}^{u_i}}\right)$, and its range is [0,1]. Generally, the closer an activity is to the current time, the greater the influence of the activity on the current interest of the user. Thus, $\mathrm{R}\left(\mathrm{e_{t_j}^{u_i}}\right)$ can be calculated by Eq. (3-9).

$$R\left(e_{t_j}^{u_i}\right) = \frac{t_j - t_{initial}^{u_i}}{t_{over}^{u_i} - t_{initial}^{u_i}}, \tag{3-9}$$

where t_j is the occurring time of an activity, $t_{initial}^{u_i}$ is the earliest time for user u_i to answer in the community, and $t_{over}^{u_i}$ is the latest time of the response behaviour of the user in the community. For the answering behaviour, the earlier it occurs, the lower its influence on the potential interest of the current user is. The closer the answering behaviour occurs at the current time, the greater its influence.

To further highlight the importance of recent activities, the definition of answering activities importance is introduced as the following.

Definition 2. Answering Activities Importance: The closer the time of an answering activity occurs at the current time, the smaller the influence. It is denoted as $\mathrm{p}\left(\mathrm{e_{t_j}^{u_i}}\right)$. Based on the logistic function, $\mathrm{p}\left(\mathrm{e_{t_j}^{u_i}}\right)$ is calculated by Eq. (3-10).

$$p\left(e_{t_j}^{u_i}\right) = \frac{1}{1 + e^{-\left(R\left(e_{t_j}^{u_i}\right) \times o\right)}}, \tag{3-10}$$

where o is the adjustment factor. Considering that the range of $R\left(e_{t_j}^{u_i}\right)$ is [0, 1], the range of $p\left(e_{t_j}^{u_i}\right)$ is [0.5, 1]. This importance can not only highlight the influence of current activities but also retain the effect of habitual activities on the current potential interests of the users.

Therefore, the importance collections of each answering activity of user u_i is $E_{u_i} = \left(p\left(e_{t_1}^{u_i}\right), p\left(e_{t_2}^{u_i}\right), \cdots, p\left(e_{t_n}^{u_i}\right)\right)$. After combining it to the topic distribution $Z^{(a^n)} = \left(z_1^{(a^n)}, z_2^{(a^n)}, \cdots, z_K^{(a^n)}\right)$ of the user's answers extracted by LC-LDA in Section 3.1.1, the topic representation of the user domain coupling interest drift can be calculated by Eq. (3-11).

$$z_k^{(u_i)} = \sum_{n=1}^{l} z_k^{(a_n)} \times p\left(e_{t_n}^{u_i}\right), \tag{3-11}$$

where l is the total times that user u_i answered in CQA, a_n is the n-th answer record of the user, $z_k^{(a^n)}$ indicates the probability that a_n belongs to the k-th topic, $p\left(e_{t_n}^{u_i}\right)$ is the importance of a_n, and $z_k^{(u_i)}$ represents the probability that u_i belongs to the k-th topic. Thus, the topic representation of the user domain integrated into the interest drift is $Z^{(u_i)} = \left(z_1^{(u_i)}, z_2^{(u_i)}, \cdots, z_K^{(u_i)}\right)$.

Similarly, combining the importance collections $E_{u_i} = \left(p\left(e_{t_1}^{u_i}\right), p\left(e_{t_2}^{u_i}\right), \cdots, p\left(e_{t_n}^{u_i}\right)\right)$ of all answering behaviors of user u_i with the fine-grained semantic representation $v^{(a^n)} = \left(x_1^{(a^n)}, x_2^{(a^n)}, \cdots, x_N^{(a^n)}\right)$ obtained by BERT in Section 3.1.1, the fine-grained semantic representation $v^{(u_i)} = \left(x_1^{(u_i)}, x_2^{(u_i)}, \cdots, x_N^{(u_i)}\right)$ of the user domain optimized by the interest drift is

$$x_n^{(u_i)} = \sum_{m=1}^{l} x_n^{(a_m)} \times p\left(e_{t_m}^{u_i}\right), \tag{3-12}$$

where l is the total number of responses provided by user u_i in CQA, a_m is the m-th answer record of this user, $x_n^{(a_m)}$ represents the value of the n-th dimension semantic vector of a_m, $p\left(e_{t_m}^{u_i}\right)$ is the importance of a_m, and $x_n^{u_i}$ is the n-th dimension semantic vector of domain representation of u_i.

The above method can provide the topic representation $Z^{(u_i)} = \left(z_1^{(u_i)}, z_2^{(u_i)}, \cdots, z_K^{(u_i)}\right)$ and the fine-grained semantic representation $v^{(u_i)} = \left(x_1^{(u_i)}, x_2^{(u_i)}, \cdots, x_N^{(u_i)}\right)$ corrected by user interest drift. Then cosine similarity is calculated to measure the domain matching degree between the question and user, shown in Eq. (3-13).

$$Sim_{q_j,u_i}^{c} = \frac{\sum_{k=1}^{K} vec_{q_j,k}^{c} \cdot vec_{u_i',k}^{c}}{\sqrt{\sum_{k=1}^{K}\left(vec_{q_j,k}^{c}\right)^2}\sqrt{\sum_{k=1}^{K}\left(vec_{u_i,k}^{c}\right)^2}}, \tag{3-13}$$

where c represents the type of semantic vectors, including the topic representation or the fine-grained representation; K is the dimension of the c_{type} semantic vector; $vec_{q_j,k}^{c}$ is the value of the k-th dimension of question q_j in the c_{type} semantic representation; $vec_{u_i',k}^{c}$ is the value of the k-th dimension of user u_i in the c_{type} semantic representation; and Sim_{q_j,u_i}^{c} is the matching degree between question q_j and user u_i in the c_{type} semantic representation.

Using Eq. (3-13), similarity Sim_{q_j,u_i}^{LC-LDA} can be calculated through the topic representation of user u_i and question q_j, and similarity Sim_{q_j,u_i}^{BERT} can be calculated through the fine-grained semantic representation of user u_i and the topic representation of question q_j. The calculation of matching degree between the question and user is shown in Eq. (3-14).

$$Sim_{q_j,u_i} = \delta Sim_{q_j,u_i}^{LC-LDA} + (1-\delta) Sim_{q_j,u_i}^{BERT}, \tag{3-14}$$

where δ is the adjustment factor that can provide the domain matching degree $\left\{Sim_{q_j,u_1}, Sim_{q_j,u_2}, \cdots, Sim_{q_j,u_M}\right\}$ between question q_j and each user in CQA. After ranking the domain matching degree of all expert users, the top K experts $\{u_{sim_1}, u_{sim_2}, \cdots, u_{sim_K}\}$ related to the domain with potential interest in question q_j can be identified.

3.2. Expert ranking based on quality optimizing TSWPR

The quality of users often determines the quality of questions in CQA, and the questions queried by higher-quality users are often more representative. Therefore, it is necessary to incorporate user quality factors into the process of expert finding. In Section 3.1, domain experts who are interested in the question are searched for at first. Based on the quality optimizing TSWPR, quality factors are integrated into the process of professional level extraction. Additionally, a collection of domain expert users with potential interest to the question contributes to a high professional level, and a high quality can be obtained at the end.

3.2.1. Construction of user quality model

The honour system in CQA provides the honour evaluation information of the users, such as the number of views, the number of likes, and the reputation score provided by CQA for each user, which can give a reference to the quality of users.

The number of views that a user has refers to the times that the questions and answers of the user have been viewed. It is known that only valuable contents can attract more users to browse in CQA. Therefore, the following assumption is made.

Hypothesis 2. There is a positive correlation between the user quality and the number of views that a user gained. A user who has received more views implies that the user is at a higher professional level.

The number of likes a user receives refers to the times the questions and answers of the user have been liked. This index can directly reflect the value of the content published by this user, and often only high-quality content can be liked by other users. Therefore, the following assumption is made.

Hypothesis 3. There is a positive correlation between the user quality and the number of likes that a user gained. A user who gains more likes implies that the user is at a higher quality level.

The user honour score provided by the CQA honour system can evaluate users from the perspective of the community value and reflect the user quality to a certain extent. Therefore, the following assumption is made.

Hypothesis 4. There is a positive correlation between the user quality and the user honour score. A higher user honour score implies a higher user quality.

In summary, the views number num_{view}^{user}, the likes number num_{like}^{user} and the honour score num_{rep}^{user} of a user are comprehensively considered when constructing the user quality model in this study. However, their dimensions are different. If these evaluation indicators of different dimensions are directly used to construct a user quality model, it will affect the accuracy of the quality model. Therefore, the maximum minimum normalization method is used to unify the dimensions of these three indices by Eq. (3-15).

$$x' = \frac{x - X_{min}}{X_{max} - X_{min}}, \tag{3-15}$$

where x is the original value, X_{max} and X_{min} represent the maximum and minimum of the values set respectively, and x' represents the normalized value. Furthermore, a user quality model is proposed according to Eq. (3-16).

$$quality_{user} = \frac{num'^{user}_{like} + num'^{user}_{view} + num'^{user}_{rep}}{3}, \tag{3-16}$$

where $quality_{user}$ is the quality of $user$, num'^{user}_{like}, num'^{user}_{view} and num'^{user}_{rep} are the normalized values of num_{view}^{user}, num_{like}^{user} and num_{rep}^{user} respectively. Thus, the quality collections $quality_{USER} = \left\{quality_{user_1}, quality_{user_2}, \cdots, quality_{user_L}\right\}$ of all users in CQA can be obtained.

3.2.2. Professional level extraction incorporating user quality

In order to extract the professional level of the users, the quality optimizing TSWPR algorithm [26] is employed. It is necessary to construct a directed interactive network of the users in CQA, which can directly reflect both the question-answer relationship among users and the effectiveness of the behaviour of each user. The user question answering interactive network is defined as follows.

Definition 3. User Question Answering Interactive Network: It is a directed graph model based on users' question answering records of the users in CQA; it is shown in Fig. 3. as an example and represented by Eq. (3-17).

$$G = (U, E), \tag{3-17}$$

where U is the set of all users in CQA, including both questioners and answerers. E represents the set of edges in the graph, and one of the directed edges $e_{ij} = (u_i, u_j) \in E$indicates that user u_j has answered the question asked by u_i. There is a weight $w_z(i \to j)$ for each directed edge. It represents the probability that user u_j will answer questions of user u_i on topic z, and the calculation formula of the weight determines the effectiveness of the final extracted professional level.

In order to filter out "indiscriminate" users, TSWPR uses Eqs. (3-18) and (3-19) to calculate $w_z(i \to j)$.

$$w_z(i \to j) = \frac{\sum_{q \in Q(i) \cap A(j)} \left(Sim^{LC-LDA}_{Z\left((u_i)_q\right), Z\left((u_i)_a\right)} \cdot s_j + 1 \right)}{\sum_{t: u_i \to u_t} \sum_{q \in Q(i) \cap A(t)} \left(Sim^{LC-LDA}_{Z\left((u_i)_q\right), Z\left((u_i)_a\right)} \cdot s_t + 1 \right)}, \tag{3-18}$$

$$s = \begin{cases} score, & score > 0 \\ 1, & score = 0 \\ 0, & score < 0 \\ 2 \cdot score, & Accepted \end{cases}, \tag{3-19}$$

where $Sim^{LC-LDA}_{Z^{((u_i)_q)},Z^{((u_i)_a)}}$ represents the similarity of topic z between question q asked by user u_i and answer a provided by user u_j. It can be calculated based on Eq. (3-14) and the topic representation of q and a extracted by LC-LDA. *score* is the score of the answer provided by user u_j, *Accepted* represents that the answer provided by user u_j is the accepted answer to question q. By incorporating information such as question acceptance, answer scores, and topic similarity into the weight calculation of the edges in the directed graph, TSWPR can effectively identify and filter "indiscriminate" users.

Eq. (3-19) can be further improved by introducing both the modified domain representation $Z^{(u_i)} = \left(z_1^{(u_i)}, z_2^{(u_i)}, \cdots, z_K^{(u_i)}\right)$ obtained in Section 3.2 and the user quality $quality_{USER} = \left\{quality_{user_1}, quality_{user_2}, \cdots, quality_{user_L}\right\}$ obtained in Section 3.3.1, so that the weight of edge can incorporate user interest drift and quality factors. This is shown in Eq. (3-20).

$$w_z(i \rightarrow j) = \frac{\sum_{q \in Q(i) \cap A(j)} \left(\left(Sim_{Z^{((u_i)_q)},Z^{((u_i)_a)}} + quality_{u_j}\right) \cdot s_j + 1\right)}{\sum_{t:u_i \rightarrow u_t} \sum_{q \in Q(i) \cap A(t)} \left(\left(Sim_{Z^{((u_i)_q)},Z^{((u_i)_a)}} + quality_{u_t}\right) \cdot s_t + 1\right)}, \tag{3-20}$$

where $quality_{u_j}$ is the quality of user u_j, and $Sim_{Z^{((u_i)_q)},Z^{((u_i)_q)}}$ is the similarity between the corrected domain representation of user u_i and u_j on topic z. By introducing quality evaluation and interest drift, the weight can be used to comprehensively evaluate the quality among the interaction of the users.

Using the above method, a directed graph of the user interactive network is constructed by integrating user quality and interest drift. The quality optimized TSWPR is then used to obtain the professional level of a user, as shown in Eq. (3-21).

$$R_{u_i}^{(z)} = \lambda \sum_{j:u_j \rightarrow u_i} R_{u_j}^{(z)} \cdot w_z(j \rightarrow i) + (1-\lambda) p_z(u_i), \tag{3-21}$$

where $\lambda \in (0,1)$ is the damping factor, which means that user u_i has a probability of $(1-\lambda)p_z(u_i)$ to randomly answer a question of another user on topic z, and $R_{u_i}^{(z)}$ represents the professional level of user u_i on topic z. With this method, the professional level collections $R_{USER}^{q_j} = \left\{R_{u_1}^{(z)}, R_{u_2}^{(z)}, \ldots, R_{u_M}^{(z)}\right\}$ of question q_j on topic z can be achieved.

3.2.3. Domain expert ranking

Sections 3.1 and 3.2.2 described how to semantically obtain domain experts and incorporate quality factors into the evaluation of the professional level. This section will integrate field matching degree and professional level to find high-quality interested experts.

Specifically, based on the expert domain representation module and the quality optimized TSWPR algorithm, the domain matching degree $\left\{Sim_{q_j,u_1}, Sim_{q_j,u_2}, \cdots, Sim_{q_j,u_M}\right\}$ and the professional level $R_{USER}^{q_j} = \left\{R_{u_1}^{(z)}, R_{u_2}^{(z)}, \cdots, R_{u_M}^{(z)}\right\}$ of all users on question q_j are obtained. Subsequently, the domain matching degree and professional level of user u_i are integrated through Eq. (3-22).

$$Score_{q_j,u_i} = \xi \cdot Sim_{q_j,u_i} + (1-\xi) \cdot R_{u_i}^{(z)}, \tag{3-22}$$

where ξ is adjustment factor, and the final value needs to be determined through experiments. $Score_{q_j,u_i}$ is the final matching score of question q_j and user u_i. It can result in the matching degree collections $\left\{Score_{q_j,u_1}, Score_{q_j,u_2}, \cdots, Score_{q_j,u_M}\right\}$ corresponding to experts who match with question q_j. Finally, experts can be ranked according to the matching degree collections, achieving the collections $Expert_K^{q_j} = \{u_{Score_1}, u_{Score_2}, \cdots, u_{Score_K}\}$ of K most relevant experts. This is a collection of experts with potential interests, similar semantic representations at multi-granularity, high evaluation quality, and high professional level. It can effectively ensure that the questions are answered by experts.

4. Experiments

4.1. Experimental settings

4.1.1. Dataset and pre-processing

In order to evaluate the effectiveness of this proposed framework for expert finding, a series of comparative experiments were conducted on two datasets of two real-world CQA websites: the Android dataset of Stack Overflow, published by Hoogeveenet al. [24] and the Program dataset of Stack Exchange [31]. Each dataset contains all questions and all users' historical Q&A records. The datasets are mutually exclusive in topics so that can comprehensively evaluate the effect of HQExpert. Other CQA datasets, such as Yahoo! Answers, are not selected for evaluation since they do not have “accepted” answers that serve as the ground truth.

The datasets are composed of those records of users’ interaction from July 20, 2009 to September 14, 2014. The details of the experimental datasets are listed in Table 1.

In order to verify the proposed method effectively, the following pre-processing was carried out on the datasets.

- The users who have never answered any questions are deleted from the answer sets.
- The users without honour value or enough personal information are not included in the user sets.
- The questions whose accepted answers id does not exist in the answer table are not included in the question sets.

This leads to 1889 questions and 2200 users in the Android dataset and 1512 questions and 3100 users in the Program dataset.

4.1.2. Evaluation metrics

The accuracy rate (ACC@N) and the mean reciprocal rank (MRR@N) were adopted to measure the effectiveness of the proposed expert finding method.

(1) ACC@N

ACC@N is the ratio between the number of answerers hit by the recommendation list and the length of the recommendation list. ACC@N is computed by Eqs. (4-1) and (4-2).

$$ACC@N = \frac{1}{|Q|}\sum\nolimits_{q \in validation} s_q, \tag{4-1}$$

$$s_q = \begin{cases} 1 & if \quad accept \in Expert^q_{Score_N} \\ 0 & if \quad accept \notin Expert^q_{Socre_N} \end{cases}, \tag{4-2}$$

where $|Q|$ is the number of retrieval questions; s_q represents the score of one retrieval of question q; and N is set to 1, 5, 10, and 15. It is evident that a higher ACC@N implies a better effectiveness for expert finding.

(2) MRR@N

MRR@N is a statistical metric for information retrieval to evaluate a list of possible similar problems generated by a question, and the problems are sorted by their probabilities of correctness. This evaluation metric is used to evaluate the effectiveness of the proposed method. The metric considers not only the hit rate of the top N retrieval results but also the ranking information of the correct results. The MRR@N is calculated by Eq. (4-3).

$$MRR@N = \frac{1}{|Q|}\sum\nolimits_{q \in Q} \frac{1}{Rank_q^{accept}}, \tag{4-3}$$

where $Rank_q^{accept}$ is the ranking of the standard answers to the top N questions retrieved by the expert finding method for question q. N equals to 1, 5, 10, and 15. Thus, their MRR expressions are MRR@1, MRR@5, MRR@10, and MRR@15.

4.1.3. Competing methods

We employed the following six methods as baselines in our experiments, including topic-based methods, authority-based methods, a combination of both and heterogeneous network-based method.

- ExpertsRank algorithm [29]: ExpertsRank utilizes asker-answerer relation to construct users’ network, and then exploits link structure analysis based on PageRank to find experts in CQA.
- Matrix Factorization (MF) algorithm [9]: The MF algorithm is a standard rating prediction model that only uses ratings for collaborative filtering. In CQA, by constructing the score matrix of the questions and the answerers, the problem is solved according to the predicted score.

Table 1
Statistics of the datasets.

Dataset	# of questions	# of comments	# of answers	# of users
Android	23,264	201,981	29,448	34,424
Program	31,034	308,695	21,417	70,253

- TSPM algorithm [8]: The TSPM algorithm is a topic-sensitive probabilistic method for expert finding in CQA systems, which fits an LDA-based probabilistic model to the question-answering activities.
- L-LDA based method [11]: Similar to TSPM, this method firstly extracts the topics of a question and user domain, and then completes expert matching. By comparing it with TSPM, the effectiveness of the supervised probabilistic topic model can be validated.
- CRAR algorithm [19]: The CRAR algorithm ranks user authority based on link analysis based on both the target question category and its relevant categories for expert finding using topic model.
- NeRank [31]: The NeRank model jointly learns the representation of question, raisers and answerers via a heterogeneous network embedding algorithm based on LSTM and metapath2vec[40].

The method of HQExpert proposed in this study consists of two parts: the expert domain representation module which is based on multi-granularity semantic analysis and interest drift, and the expert ranking module (RKExpert) which is based on quality optimization TWSPR. In order to verify the effectiveness of the two parts, we performd comparison experiments between HQExpert and a few derived internal models to prove the efficacy of its design, respectively.

- LC-LDA based expert finding method (LC-EF): This method uses LC-LDA to extract topics from question and user domain, and then achieve the matching of experts.
- BERT based expert finding method (BERT-EF): This method uses BERT to extract semantic vectors from question and user domain, and then achieve expert matching.
- Expert finding method based on the coupled LC-LDA and BERT (LCBT-EF): This method combines LC-EF and BERT-EF, using the similarity between the question and the user domain calculated by these two methods to find the domain experts. By comparing LC-LDA and BERT, the effectiveness of the coupled LC-LDA and BERT method can be verified.
- Interest improved LC-LDA based expert finding method (IMLC-EF): Interest drift is introduced based on LC-EF, and the LC-LDA based topic representation of expert domain is modified to verify the effectiveness of interest drift factor.
- Interest improved BERT based expert finding method (IMBT-EF): Interest drift is introduced based on BERT-EF, and the BERT based fine-grained semantic representation of expert domain is modified to verify the effectiveness of the interest drift factor.
- Interest improved expert finding method coupled with LC-LDA and BERT (IMLCBT-EF): The domain expert finding method proposed in Section 3.1 can verify the impact of the interest drift factor.
- TSWPR based expert finding method coupled with LC-LDA and BERT (TSIMLCBT-EF): Based on IMLCBT-EF, the TSWPR algorithm is used to extract the professional level of the user. The results of the domain expert matching are modified to verify the impact of expertise.
- HQExpert is compared to TSIMLCBT-EF to verify the effectiveness of the expert finding method incorporating user quality.

4.1.4. Parameter setting

HQExpert uses the following parameters: the expansion coefficient r in Eq. (3-3), the Dirichlet hyperparameters α and β of the LC-LDA model, the adjustment factor δ in Eq. (3-14), the damping factor λ in Eq. (3-21), and another adjustment factor ξ in Eq. (3-22). The expansion coefficient r was set to 2.0 in the next experiments [26]; the Dirichlet hyperparameters were set as α = 0.5 and β = 0.1, according to Liu et al. [22] and the damping factor was set to λ = 0.2 according to Liu et al. [26] The sensitivity of the two important adjustment factors, i.e., δ in Eq. (3-14) and ξ in Eq. (3-22), will be studied in Section 4.2.1 through experiments.

4.2. Experimental results and analysis

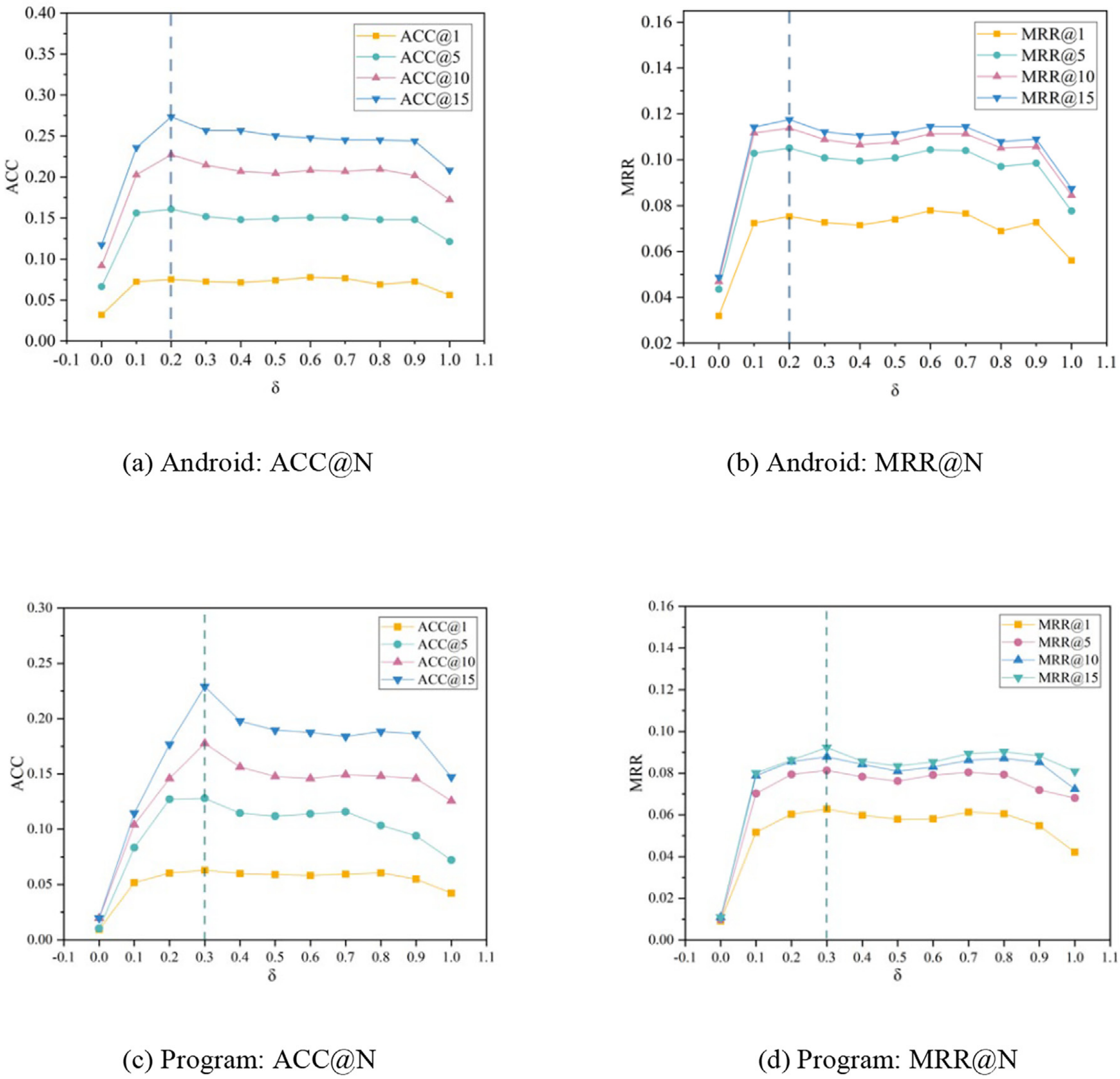
4.2.1. Parameter sensitivity

δ in Eq. (3-14) decides the proportion of coarse-grained and fine-grained semantics in domain matching. In order to determine this parameter, coarse-grained and fine-grained semantic similarities were extracted respectively, and domain experts were matched with intervals of 0.1 in the range of [0,1]. In order to evaluate the sensitivity of δ, ACC@N and MRR@N were used as metrics, and Fig. 4 shows the results of the parameter sensitivity analysis.

From the results, we found that the performance continuously improved when the proportion of coarse-grained semantics obtained by LC-LDA increased until a turning point. If the proportion of coarse-grained semantics was too large, the performance started to degrade since it was easy for the semantic representations to be generalized and ignore the deeper meaning implied in questions. For instance, the performance of the dataset Android started to degrade after δ was larger than 0.2. Therefore, the parameter δ was set to 0.2 in the following experiments.

Parameter ξ decides the ratio of domain matching factor and expertise factor in the retrieval of high-quality experts. To determine the optimal value of ξ, ACC@N and MRR@N were used as evaluation metrics after extracting the domain matching degree and professional level of the user. Fig. 5 shows the parameter sensitivity on ξ in the Android and Program datasets.

It is observed that the curves have similar transformation trend in the Fig. 5, meaning the speed reaching the optimum value for the parameter ξ is similar in different datasets. However, we observe that the optimal value is reached slightly slower in Program dataset than in the Android dataset. For instance, when the parameter ξ is 0.05, the performance of ACC@N reaches the highest in Android dataset. However, in Program dataset, when the parameter ξ is 0.2, the accuracy is



(a) Android: ACC@N

(b) Android: MRR@N

(c) Program: ACC@N

(d) Program: MRR@N

Fig. 4. Parameter Sensitivity Analysis on δ.

the largest. Owing to the diversity of Android questions being less than that of the program questions, a high proportion of domain representations were needed to supplement.

4.2.2. Effectiveness of interest drift model and user quality model

The user's interest preferences in different periods are not static, and interest drift will impact the results of expert finding, leading to questions being recommended to uninterested experts. In order to solve this problem, this study proposes a calculation formula of interest drift. To verify whether the model can effectively characterize user's interest drift, the activity records of the user with ID "10642" in Stack Overflow are taken as an example to analyze and calculate this user's interest drift. Its results are shown in Table 2.

This user answered a total of 8 questions from December 2011 to May 2014. Before 2014, answers involved questions mainly about the applications in Android system, which mainly included "networking", "USB-drivers" and "Skype". For these activities, interest drift degrees were lower than 0.9. However, after 2014, the interest of this user changed and gradually turned to the kernel of the Android system, including "cache", "internal-storage", and "instagram". Thus, the interest drift degrees of these activities were higher; all were above 0.95. It can be seen that this user has a long-term interest in knowledge related to the "usb", and this was retained in the process of describing the interest distribution.

Furthermore, the user quality is defined and characterized in the expert ranking process, and its statistical distribution is shown in Fig. 6. It can be seen that users in CQA are distributed with a long tail. More than 60% of users have relatively low quality, and their quality values are between [0, 0.004], while users with higher quality only account for less than 40%, and

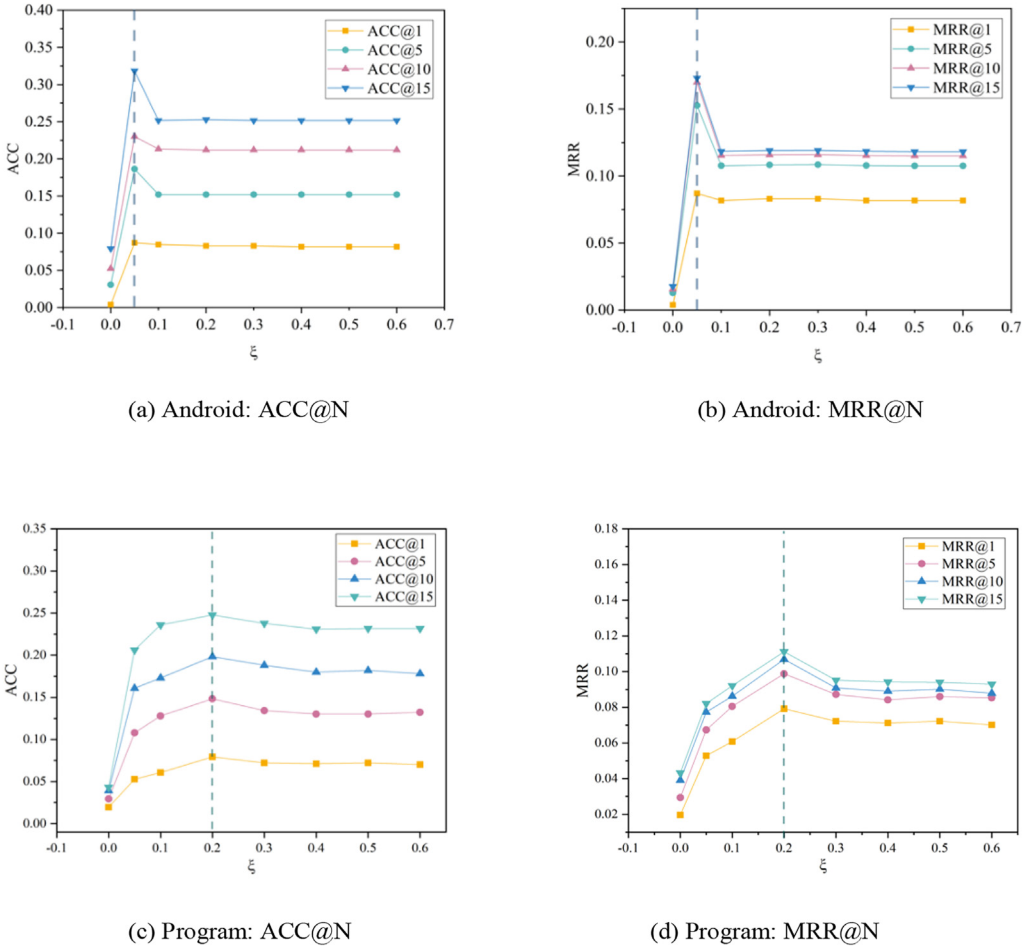


Fig. 5. Parameter Sensitivity Analysis on ξ.

Table 2
Activity record and interest drift of user "10642" in Stack Overflow. There are eight answer records, which contain the answer ID, answer time, interest drift, and answer label.

Answer ID	Answer time	Interest drift	Answer label
16999	2011-12-15	0.5	['skype']
19088	2012-2-3	0.57473	['2.3-gingerbread', 'networking', 'streaming']
27755	2012-8-16	0.81342	['usb', 'samsung-nexus-s', 'adb', 'usb-drivers']
55289	2013-10-17	0.98261	['applications', 'downloading']
67463	2014-4-13	0.99395	['editing', 'android-tv']
69578	2014-5-18	0.99508	['usb', 'charging']
69591	2014-5-18	0.99509	['cache', 'instagram']
82316	2014-9-10	0.99753	['internal-storage', 'nexus-4', 'music']

the number of users whose quality is higher than 0.06 accounts for approximately 7%. In summary, this distribution is in line with the professional level distribution of the users in CQA, that is, there are a large number of average users and a small number of expert users.

4.2.3. Comparison of different baseline models

In this subsection, we present the comparison results of the benchmark models to verify the effectiveness of HQExpert. Six baseline models (i.e., ExpertsRank [29], MF [9], TSPM [8], L-LDA [11], CRAR [19], and NeRank [31]) were used to conduct experimental comparisons of HQExpert. HQExpert significantly outperformed all baseline models on both datasets in terms of all the metrics, and several observations stand out from Table 3.

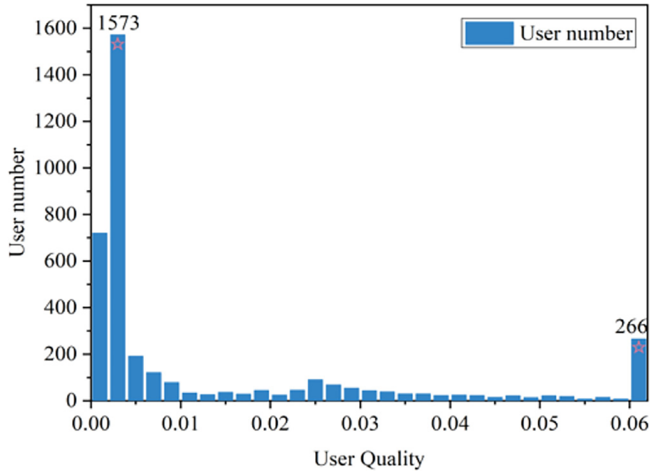


Fig. 6. User quality distribution in CQA. The pink stars represent the number of users whose quality factors are between [0.002, 0.004] and above 0.06. Most users are not experts, and only 266 users have a quality factor greater than 0.06. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 3
Comparisons of ACC@N and MRR@ between HQExpert and the six baseline models.

Dataset	Metric	ExpertsRank	MF	TSPM	L-LDA	CRAR	NeRank	HQExpert
Android	ACC@1	0.0278	0.0332	0.0416	0.0467	0.0715	0.0801	**0.0870**
	ACC@5	0.054	0.0693	0.1372	0.1231	0.1532	0.1792	**0.1864**
	ACC@10	0.0818	0.1067	0.1469	0.1521	0.2015	0.2312	**0.2410**
	ACC@15	0.1067	0.1314	0.1852	0.2012	0.2421	0.2921	**0.3183**
	MRR@1	0.0278	0.0332	0.0416	0.0516	0.0715	0.0801	**0.0871**
	MRR@5	0.0402	0.0584	0.0737	0.0737	0.1041	0.1401	**0.1526**
	MRR@10	0.0441	0.0639	0.0789	0.0789	0.1101	0.1477	**0.1699**
	MRR@15	0.0485	0.0664	0.0812	0.0812	0.1167	0.1523	**0.1730**
Program	ACC@1	0.0196	0.0294	0.0392	0.0432	0.0629	0.0734	**0.0792**
	ACC@5	0.0686	0.0784	0.1034	0.1144	0.1278	0.1321	**0.1482**
	ACC@10	0.0686	0.0882	0.1313	0.1383	0.1672	0.1792	**0.1982**
	ACC@15	0.0882	0.098	0.1721	0.1801	0.2162	0.2216	**0.2477**
	MRR@1	0.0196	0.0294	0.0392	0.0432	0.0629	0.0734	**0.0792**
	MRR@5	0.0392	0.0493	0.0542	0.0592	0.0836	0.0901	**0.0988**
	MRR@10	0.0392	0.0503	0.0553	0.0623	0.0876	0.0924	**0.1069**
	MRR@15	0.0407	0.051	0.0561	0.0625	0.0913	0.0945	**0.1111**

- From the results shown in Table 3, it can be seen that ExpertsRank performs worst among all methods on both datasets. This is because ExpertsRank uses link structure analysis based on the PageRank algorithm which only considers the professional level of answerers in the network. The result proves link structure-based methods cannot work well in the CQA expert finding scenario.
- MF performs better on different evaluation metrics than ExpertsRank, indicating that it is more reliable to obtain the professional degree according to the question than only according to the link structure.
- TSPM, a topic-sensitive probabilistic method, had a 4.25% improvement on ACC@N, 1.58% improvement on MRR@N on the Android dataset and a 2.45% improvement on the ACC@N on the Program dataset when compared with MF, which shows that compared with link structure information, topic matching has a greater impact on the correctness of the results.
- The supervised probabilistic topic model, L-LDA, outperforms TSPM by 0.3% to 0.75% on MRR@N, which suggests that using the tags in CQA as the labels for the probabilistic topic model can better represent the domain information of the experts.
- The results of CRAR are improved by 1.97% on MRR@1 on the Program dataset and 2.92% on MRR@N on the Android dataset when compared to L-LDA, which proves the effectiveness of combining topic features with link structure to improve expert finding. Overall, combined methods obtain better performances than single.
- NeRank performs best out of all the baselines. This is because NeRank uses LSTM to extract the specific context information in the sentences and exploits the metapath-based network embedding algorithm to learn the representations of question, raisers, and answerers.

- Finally, compared to NeRank, HQExpert records a 1.25% improvement for ACC@N on the Android dataset and a 1.66% improvement for MRR@5 on the Program dataset. We attribute the superiority of HQExpert to its three properties: 1) For learning the experts' representations more comprehensively, HQExpert considers not only coarse-grained topic information but also fine-grained sentence information based on a stronger pre-training model; 2) In addition to the multi-granularity domain representations, which is based on the content, the impact of the interest shift is also taken into account to generate time-sensitive domain representations; 3) For measuring the level of the experts, HQExpert incorporates user quality on the basis of the professional level extraction method TWSPR.

4.2.4. Comparison of HQExpert variants

To verify the effectiveness of each part of the HQExpert method, a few comparison experiments were conducted between HQExpert and other derived models on different datasets. Experimental results are shown in Table 4 and Fig. 7 for the Android dataset, and Table 5 and Fig. 8 for the Program dataset. Firstly, by comparing LC-EF and BERT-EF, it can be found that the former achieves better results than BERT-EF and IMBT-EF on both datasets for all metrics. The reason is that LC-LDA was trained on a relatively small amount of data set that could effectively represent text semantics. Whereas BERT trained with massive data in this comparison experiment. Secondly, by comparing LCBT-EF with BERT-EF, it can be found

Table 4
Experimental results of HQExpert and derived models on Android dataset.

Model	ACC@1	ACC@5	ACC@10	ACC@15	MRR@1	MRR@5	MRR@10	MRR@15
LC-EF	0.0561	0.1213	0.1724	0.2081	0.0561	0.0777	0.0845	0.0874
BERT-EF	0.0319	0.0664	0.0919	0.1174	0.0319	0.0435	0.0469	0.0488
LCBT-EF	0.0766	0.1507	0.2069	0.2452	0.0766	0.1040	0.1113	0.1144
IMLC-EF	0.0587	0.1238	0.1749	0.2107	0.0587	0.0805	0.0874	0.0903
IMBT-EF	0.0306	0.0673	0.0989	0.1213	0.0306	0.0497	0.0538	0.0544
IMLCBT-EF	0.0817	0.1519	0.2120	0.2490	0.0817	0.1076	0.1150	0.1180
TSIMLCBT-EF	0.0830	0.1502	0.2107	0.2615	0.0830	0.1287	0.1358	0.1391
HQExpert	**0.0870**	**0.1864**	**0.2300**	**0.3183**	**0.0870**	**0.1526**	**0.1699**	**0.1730**

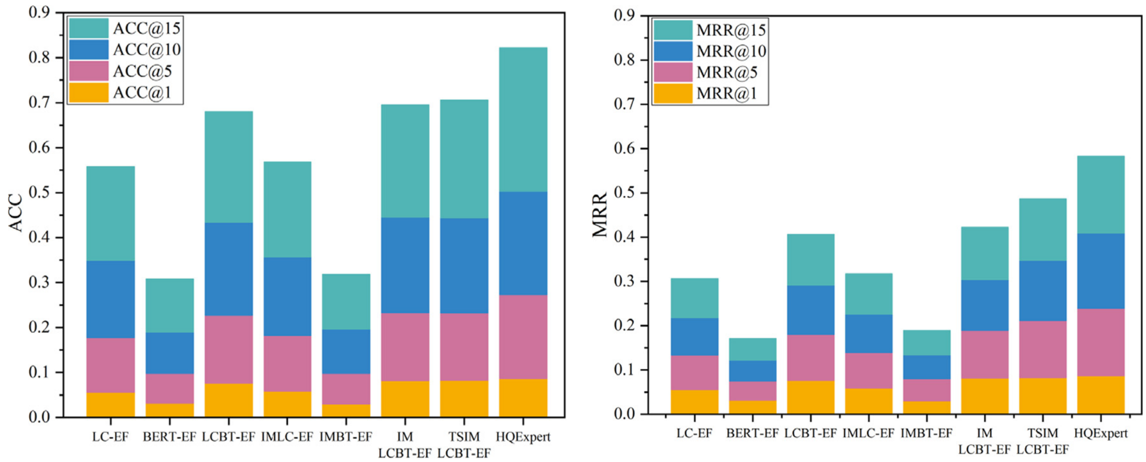


Fig. 7. ACC@N (left) and MRR@N (right) of HQExpert and derived models on Android dataset.

Table 5
Experimental results of HQExpert and derived models on Program dataset.

Model	ACC@1	ACC@5	ACC@10	ACC@15	MRR@1	MRR@5	MRR@10	MRR@15
LC-EF	0.0421	0.0721	0.1256	0.1473	0.0421	0.0681	0.0724	0.0809
BERT-EF	0.0093	0.0103	0.0198	0.0198	0.0093	0.0099	0.011	0.011
LCBT-EF	0.0628	0.1258	0.1776	0.2011	0.0628	0.0813	0.0879	0.0923
IMLC-EF	0.0495	0.0688	0.1383	0.1529	0.0495	0.0733	0.0799	0.0828
IMBT-EF	0.0116	0.0123	0.0233	0.0318	0.0116	0.0116	0.0129	0.0129
IMLCBT-EF	0.0698	0.1279	0.1808	0.2116	0.0698	0.0934	0.0981	0.1016
TSIMLCBT-EF	0.0732	0.1348	0.1891	0.2208	0.0732	0.0952	0.1021	0.1052
HQExpert	**0.0792**	**0.1482**	**0.1982**	**0.2477**	**0.0792**	**0.0988**	**0.1069**	**0.1111**

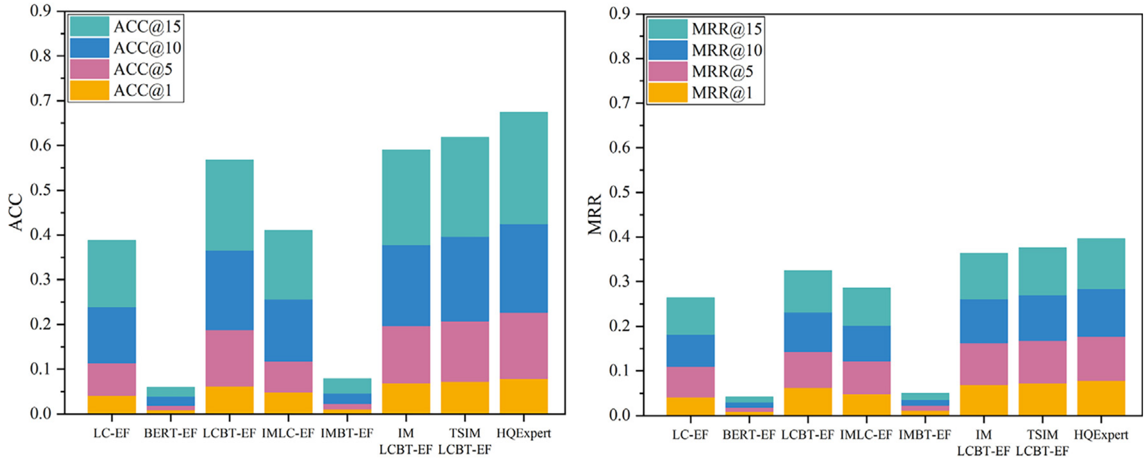


Fig. 8. ACC@N (left) and MRR@N (right) of HQExpert and derived models on Program dataset.

that the former clearly improved, with an average increase of 9.29% for ACC@N on the Android dataset and 12.7% improvement for ACC@N on the Program dataset. Compared with LC-EF, LCBT-EF still achieved better results. This shows that multi-granularity semantical representation (LCBT-EF) performs better than single-granularity methods. IMLCBT-EF is better than LCBT-EF without the interest shift model on both datasets. The results prove that the interest shift model can better capture the current preferences of the experts. Furthermore, the usage of the TSWPR algorithm in IMLCBT-EF (TSIMLCBT-EF) can improve MRR@N by 1.61% on the Android dataset and ACC@N by 0.7% on the Program dataset. This finding confirms the efficacy of using the professional level in HQExpert and shows that the TSWPR algorithm excludes indiscriminate users to a certain extent. By comparing HQExpert and TSIMLCBT-EF, the former increases by 2.91% and 1.38% on ACC@N on the two datasets, indicating that the introduction of user quality can further filter out experts with low quality.

5. Conclusion and future work

Investigating the problem of ignoring the impact of quality factors in existing expert finding methods, this study proposes a high-quality domain expert finding method in CQA based on multi-granularity semantic analysis and interest drift (HQExpert). This method consists of two parts: an expert domain representation based on multi-granularity semantic analysis and an interest drift and expert ranking strategy based on quality optimizing TSWPR (RKExpert). In expert domain representation, LC-LDA is used to extract topic representation, and fine-grained semantic representation is embedded by BERT to complete the multi-granularity semantic representation of user domain. Furthermore, the current interest domain of a user is effectively represented by extracting the potential interest distribution of the user based on activity records, and obtaining the interest drift weight of the user's activities. By using modified expert domain representation, appropriate domain experts are obtained to match the semantic representation of a question. In RKExpert, the honour value of the user in CQA (including the number of likes, the number of views, and honour scores) is used to establish a user quality evaluation model. Subsequently, a user interactive network is constructed, and the weights of the network are calculated and updated based on key information, such as user quality, question and answer scores, question acceptance, and the topic similarity between questions and answers. The quality optimizing TSWPR algorithm is used to obtain the professional level coupling with user quality of each user, thereby updating the domain expert retrieval results and then identifying high-quality domain experts with potential interest in question. Finally, the method proposed in this study is verified on the two datasets: the Android dataset and the Program dataset. Experimental results show that ACC@N of the HQExpert method has an average increase of 14.06%, 4.11%, and 1.25% compared with ExpertsRank, CRAR, and NERank on the Android dataset, respectively. Similarly, the ACC@N of HQExpert method has an average increase of 10.7%, 2.48%, and 1.6% compared with ExpertsRank, CRAR, and NERank on the Program dataset, respectively. This implies that this method can effectively retrieve high-quality experts.

In future work, this method can be explored or optimized from the following interesting research directions. Firstly, the BERT model used in this study to extract word-granularity expert semantic representation is based on a general model trained by Google, and a dedicated BERT model can be trained based on the dataset of Stack Overflow to further improve the matching accuracy. Secondly, in the HQExpert model, the coupling of LC-LDA and BERT is used to perform multi-granularity semantic analysis on the expert domain and applying it to similar question retrieval in CQA might be another interesting research direction. Finally, expert domain, interest drift, and professional level are expected to be introduced into the question response time prediction in CQA, thereby establishing a high-quality expert model to predict the time for high-quality experts to provide high-quality answers.

CRediT authorship contribution statement

Yue Liu: Conceptualization, Methodology, Validation, Formal analysis, Writing – original draft, Supervision. **Weize Tang:** Methodology, Software, Validation, Data curation, Writing – original draft, Writing – review & editing. **Zitu Liu:** Conceptualization, Methodology, Formal analysis, Writing – original draft, Writing – review & editing. **Lin Ding:** Validation, Writing – review & editing. **Aihua Tang:** Software, Formal analysis, Resources, Data curation, Writing – original draft.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work is supported by the State Key Program of National Nature Science Foundation of China (No. 61936001) and the National Natural Science Foundation of China (No. 52073169). We also thank the High Performance Computing Center of Shanghai University, and Shanghai Engineering Research Center of Intelligent Computing System (No. 19DZ2252600) for providing the computing resources and technical support.

References

[1] F. Riahi, Z. Zolaktaf, M. Shafiei, E. Milios, Finding expert users in community question answering, in: Proceedings of the 21st international conference on world wide web, 2012, pp. 791–798.
[2] B. Li, T. Jin, M.R. Lyu, I. King, B. Mak, Analyzing and predicting question quality in community question answering services, in: Proceedings of the 21st international conference on world wide web, 2012, pp. 775–782.
[3] M. Neshati, Z. Fallahnejad, H. Beigy, On dynamicity of expert finding in community question answering, Inf. Process. Manage. 53 (5) (2017) 1026–1042.
[4] A. Dargahi Nobari, S. Sotudeh Gharebagh, M. Neshati, Skill translation models in expert finding, in: Proceedings of the 40th international ACM SIGIR conference on research and development in information retrieval, 2017, pp. 1057–1060.
[5] M. Dehghan, A.A. Abin, Translations diversification for expert finding: A novel clustering-based approach, ACM Trans. Knowl. Discovery Data 13 (3) (2019) 1–20.
[6] M. Dehghan, H.A. Rahmani, A.A. Abin, V.-V. Vu, Mining shape of expertise: A novel approach based on convolutional neural network, Inf. Process. Manage. 57 (4) (2020) 102239.
[7] A.D. Nobari, M. Neshati, S.S. Gharebagh, Quality-aware skill translation models for expert finding on StackOverflow, Inf. Syst. 87 (2020) 101413.
[8] J. Guo, S. Xu, S. Bao, Y. Yu, Tapping on the potential of q&a community by recommending answer providers, in: Proceedings of the 17th ACM conference on Information and knowledge management, 2008, pp. 921–930.
[9] Y. Koren, R. Bell, C. Volinsky, Matrix factorization techniques for recommender systems, Computer 42 (8) (2009) 30–37.
[10] H. Zhu, H. Cao, H. Xiong, E. Chen, J. Tian, Towards expert finding by leveraging relevant categories in authority ranking, in: Proceedings of the 20th ACM international conference on Information and knowledge management, 2011, pp. 2221–2224.
[11] D. Ramage, D. Hall, R. Nallapati, C.D. Manning, Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora, in: Proceedings of the 2009 conference on empirical methods in natural language processing, 2009, pp. 248–256.
[12] D. Xiong, J. Wang, H. Lin, An LDA-based approach to finding similar questions for community question answer, J. Chin. Inf. Process. 26 (5) (2012) 40–45.
[13] P. Jurczyk, E. Agichtein, Discovering authorities in question answer communities by using link analysis, in: Proceedings of the sixteenth ACM conference on Conference on information and knowledge management, 2007, pp. 919–922.
[14] G. Zhou, S. Lai, K. Liu, J. Zhao, Topic-sensitive probabilistic model for expert finding in question answer communities, in: Proceedings of the 21st ACM international conference on Information and knowledge management, 2012, pp. 1662–1666.
[15] L. Page, S. Brin, R. Motwani, T. Winograd, The PageRank citation ranking: Bringing order to the web, Stanford InfoLab, 1999.
[16] J.M. Kleinberg, Authoritative sources in a hyperlinked environment, J. ACM 46 (5) (1999) 604–632.
[17] L. Yang et al, Cqarank: jointly model topics and expertise in community question answering, in: Proceedings of the 22nd ACM international conference on Information & Knowledge Management, 2013, pp. 99–108.
[18] H. Li, S. Jin, L. Shudong, A hybrid model for experts finding in community question answering, in: 2015 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery, IEEE, 2015, pp. 176–185.
[19] H. Zhu, E. Chen, H. Xiong, H. Cao, J. Tian, Ranking user authority with relevant knowledge categories for expert finding, World Wide Web 17 (5) (2014) 1081–1107.
[20] T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient estimation of word representations in vector space, arXiv preprint arXiv:1301.3781, 2013.
[21] A. Joulin, E. Grave, P. Bojanowski, T. Mikolov, Bag of tricks for efficient text classification, arXiv preprint arXiv:1607.01759, 2016.
[22] Y. Liu, A. Tang, Z. Sun, W. Tang, F. Cai, C. Wang, An integrated retrieval framework for similar questions: Word-semantic embedded label clustering–LDA with question life cycle, Inf. Sci. 537 (2020) 227–245.
[23] S. Yuan, Y.u. Zhang, J. Tang, W. Hall, J.B. Cabotà, Expert finding in community question answering: a review, Artif. Intell. Rev. 53 (2) (2020) 843–874.
[24] D. Hoogeveen, L.i. Wang, T. Baldwin, K.M. Verspoor, Web forum retrieval and text analytics: A survey, Found. Trends Inf. Retrieval 12 (1) (2018) 1–163.
[25] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, arXiv preprint arXiv:1810.04805, 2018.
[26] A. Tang, P. Ren, Z. Sun, Multi-feature based Question-Answerer Model Matching for predicting response time in CQA, Knowl.-Based Syst. 182 (2019) 104794.
[27] A. Daud, J. Li, L. Zhou, F. Muhammad, Temporal expert finding through generalized time topic modeling, Knowl.-Based Syst. 23 (6) (2010) 615–625.
[28] S. Momtazi, F. Naumann, Topic modeling for expert finding using latent Dirichlet allocation, Wiley Interdiscip. Rev. 3 (5) (2013) 346–353.
[29] J. Zhang, M.S. Ackerman, L. Adamic, Expertise networks in online communities: structure and algorithms, in: Proceedings of the 16th international conference on World Wide Web, 2007, pp. 221–230.
[30] G. Zhou, J. Zhao, T. He, W. Wu, An empirical study of topic-sensitive probabilistic model for expert finding in question answer communities, Knowl.-Based Syst. 66 (2014) 136–145.
[31] Z. Li, J.-Y. Jiang, Y. Sun, W. Wang, Personalized question routing via heterogeneous network embedding, Proceedings of the AAAI Conference on Artificial Intelligence 33 (01) (2019) 192–199.
[32] W.-C. Kao, D.-R. Liu, S.-W. Wang, Expert finding in question-answering websites: a novel hybrid approach, in: Proceedings of the 2010 ACM symposium on applied computing, 2010, pp. 867–871.

[33] S. Geerthik, K. R. Gandhi, S. Venkatraman, Domain expert ranking for finding domain authoritative users on community question answering sites, in: 2016 IEEE international conference on computational intelligence and computing research (ICCIC), IEEE, 2016, pp. 1–5.
[34] J. Sun, W. Xu, J. Ma, J. Sun, Leverage RAF to find domain experts on research social network services: A big data analytics methodology with MapReduce framework, Int. J. Prod. Econ. 165 (2015) 185–193.
[35] X.-J. Wang, X. Tu, D. Feng, L. Zhang, Ranking community answers by modeling question-answer relationships via analogical reasoning, in: Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval, 2009, pp. 179–186.
[36] R.M. Nosofsky, T.J. Palmeri, An exemplar-based random walk model of speeded classification, Psychol. Rev. 104 (2) (1997) 266.
[37] D.O. Logofet, E.V. Lesnaya, The mathematics of Markov models: what Markov chains can really predict in forest successions, Ecol. Model. 126 (2–3) (2000) 285–298.
[38] A. Vaswani et al, Attention is all you need, Advances in neural information processing systems 30 (2017).
[39] K. Cho, et al., Learning phrase representations using RNN encoder-decoder for statistical machine translation, arXiv preprint arXiv:1406.1078, 2014.
[40] Y. Dong, N.V. Chawla, A. Swami, metapath2vec: Scalable representation learning for heterogeneous networks, in: Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining, 2017, pp. 135–144.