Competition-based User Expertise Score Estimation

Jing Liu[†]*, Young-In Song[‡], Chin-Yew Lin[‡]

[†]Harbin Institute of Technology
No. 92, West Da-Zhi St, Nangang Dist.
Harbin, China 150001
jliu@ir.hit.edu.cn

[‡]Microsoft Research Asia
Building 2, No. 5 Dan Ling St, Haidian Dist.
Beijing, China 100190
{yosong, cyl}@microsoft.com

ABSTRACT

In this paper, we consider the problem of estimating the relative expertise score of users in community question and answering services (CQA). Previous approaches typically only utilize the explicit question answering relationship between askers and answerers and apply link analysis to address this problem. The implicit pairwise comparison between two users that is implied in the best answer selection is ignored. Given a question and answering thread, it's likely that the expertise score of the best answerer is higher than the asker's and all other non-best answerers'. The goal of this paper is to explore such pairwise comparisons inferred from best answer selections to estimate the relative expertise scores of users. Formally, we treat each pairwise comparison between two users as a two-player competition with one winner and one loser. Two competition models are proposed to estimate user expertise from pairwise comparisons. Using the NTCIR-8 CQA task data with 3 million questions and introducing answer quality prediction based evaluation metrics, the experimental results show that the pairwise comparison based competition model significantly outperforms link analysis based approaches (PageRank and HITS) and pointwise approaches (number of best answers and best answer ratio) for estimating the expertise of active users. Furthermore, it's shown that pairwise comparison based competition models have better discriminative power than other methods. It's also found that answer quality (best answer) is an important factor to estimate user expertise.

Categories and Subject Descriptors

H.1.2 [**User/Machine Systems**]: Human information processing; H.3.3 [**Information Search and Retrieval**]: Selection process

General Terms

Experimentation

Keywords

Expertise estimation, community question answering, pairwise comparison, competition model

1. INTRODUCTION

Search engines help people search information on the World Wide Web. However, not all human knowledge and experiences can be covered by existing web pages. With the explosive growth of web 2.0 sites, community question and answering services (denoted as CQA) such as Yahoo! Answers[1] and Baidu Zhidao[2], have become important services where people can use natural language rather than keywords to ask questions and seek advice or opinions from real people who have relevant knowledge or experiences. CQA services provide another way to satisfy a user's information needs that cannot be met by traditional search engines. Users are the unique source of knowledge in CQA sites and all users from experts to novices can generate content arbitrarily. Therefore, it is desirable to have a system that can automatically estimate the user expertise score and identify experts who can provide good quality answers. Many applications can benefit from user expertise score estimation, for example, routing questions to experts, extracting good quality answers and creating mechanisms to encourage those identified experts to participate more, etc.

Intuitively, the user expertise score can be estimated from the number of answers per user, the quality of answers, and user interaction. Several models have been proposed to estimate relative expertise scores of users in CQA, for example, analysis of the number of questions and answers [23], analysis of the number of best answers [3], link analysis [14, 23], and modeling user expertise and answer quality simultaneously [2]. Link analysis based approaches [14, 23] utilize question and answering relationships between askers and answerers to estimate the relative expertise score of users. However, answer quality, which is important for estimating user expertise, is not considered in those models. The co-training model [2], which jointly estimates user expertise and answer quality, considers answer quality, but it doesn't model user interaction explicitly. Usually, the best answer is just simply used as an individual feature to estimate user expertise score or answer quality. Despite these past efforts, there is still not a principal way to estimate user expertise score and evaluate results.

In this paper, we propose a general and simple competition-based method to estimate user expertise score. By "general", we mean that our method can be applied to all CQA services that have best answer selection. To the best of our knowledge, all existing CQA services have best answer annotation by their users. By "simple", we mean that our method assumes two simple and intuitive principles: (1) given a question answering thread, it's likely that the expertise score of the best answerer is higher than the asker; (2) Similarly, it's likely that the expertise score of the best answerer is higher than the expertise score of all the other answerers. By applying these two simple principles, we can determine relative expertise scores among users through pairwise comparison between (1) an asker and a best answerer, and between (2) a best

*This work was done when Jing Liu was a visiting student at Microsoft Research Asia.

SIGIR'11, July 24–28, 2011, Beijing, China.

[1] http://answers.yahoo.com/

[2] http://zhidao.baidu.com/

answerer and all other non-best answerers. Our goal is to explore such general and simple pairwise comparisons to estimate the relative expertise score of users. Each pairwise comparison can be viewed as a two-player competition without tie. In this paper, we present two pairwise comparison based competition models to estimate user expertise score.

For user expertise score evaluation, we use NTCIR-8 CQA task data with 3 million questions and introduce answer quality prediction based evaluation metrics to evaluate our approaches. Our main findings from experiments are that: (a) Our pairwise comparison based competition model significantly outperforms pointwise approaches, including number of answers, number of best answers and best answer ratio; and link-based approaches, including question answering relationship based PageRank and HITS; (b) the pairwise comparison based competition model shows better discriminative power for estimating the relative expertise score of users than other approaches; (c) it's also shown that, as an indicator of answer quality, the best answer is important for estimating user expertise.

This paper is structured as follows. Section 2 discusses related work on user expertise score estimation. Section 3 proposes the notion of pairwise comparisons between users and introduces the competition-based method. Section 4 presents two pairwise comparison based competition models for user expertise estimation. Section 5 introduces the answer quality prediction based evaluation metrics and evaluates the proposed methods. Section 6 concludes this paper and discusses future work.

2. RELATED WORKS

With the rapid increase of CQA sites over recent years, estimating the expertise score of CQA users has become a common task and results in a variety of approaches.

Link analysis based ranking approaches have shown its success in measuring quality of web pages. Two of the most famous link analysis approaches are PageRank [4] and HITS [15]. Early work on estimating expertise score of CQA users employs link analysis technology on the question answering relationship based user graph. In a user graph, each user is viewed as a node. If there is a question answering relationship between two users then there is a directed edge from the asker to the answerer. Zhang et al. [23] proposed the ExpertiseRank model, which is a PageRank-like algorithm, to estimate expertise score of users from online forums. Similar to PageRank, ExpertiseRank considers not only the number of users one has helped, but also whom they helped. By considering askers as hub nodes, and answerers as authority nodes, Jurczyk et al. [14] employed HITS to estimate user expertise score based on the question answering relationship between askers and answerers in CQA. However, answer quality information such as best answer, which reflects asker choices on which answer is correct, useful, readable and informative, is ignored in these works.

Due to the importance of answer quality, many following works incorporate best answer labels into user expertise score estimation. Bian et al. [2] proposed a mutual reinforcement approach for jointly modeling user expertise and answer quality. However, this work does not explicitly model user relationship. It extracted question answering relationship based link information as features. Pal et al. [18] studied user behavior and showed that experts prefer to answer questions that do not already have good answers. This is because experts recognize that they have a higher chance to make more valuable contributions to those questions. Based on this finding, they model users' question selection bias to identify experts. Typically an estimation of user expertise is presented as a ranked list of users with their expertise scores without an explicit indicator of who should be considered as experts. Bouguessa et al. [3] propose a method to solve the problem of determining how many users should be selected as experts from a user list ranked by number of best answers. They also argued that best answer is important to estimate user expertise score.

Besides the question and answering community, user expertise score estimation is also studied in other social networks. Campbell et al. [5] used HITS to compute user expertise score over the user network of e-mail communications. Zhou et al. [24] followed the PageRank paradigm to propose an approach for co-ranking authors and their publications in a heterogeneous network.

Research on user expertise score estimation can benefit a lot of applications. User expertise and other user related information are widely used for evaluating answer quality in CQA [1, 12, 17] and in online forums [7]. Suryanto et al. [22] incorporated user expertise score into question and answer search; they showed that user expertise score can improve question and answer search. Li et al. [16] proposed a framework to route questions to right answerers who are experts and available to answer questions. Horowitz et al. [11] developed a social search engine which routes questions to askers' extended social network, including Facebook and Google Contacts, rather than the question and answering community.

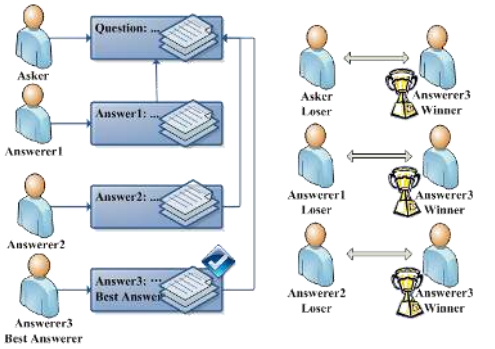
In this paper, our key idea is to propose a competition-based method that explores pairwise comparisons between users inferred from best answer selections, to estimate user expertise score. Each pairwise comparison can be treated as a two-player competition. From a competition-based perspective, expertise score estimation becomes a related problem to the calculation of the statistical skill rating of players or teams in competitive games or sports. The main research works in this area mainly studied ranking players or teams purely based on win-loss results. The most well-known skill rating system is Elo [8], which is designed to calculate the relative skill scores of players in a two-player game. It already has been widely used in many sports including chess, football, and baseball. Elo assumes that the performance of one player in a game is normally distributed around its skill level with a fixed variance and that the probability of each possible outcome of one game is determined by the skill ratings of the two players. TrueSkill [10] extends the Elo with a dynamic variance and targets one main challenge in online games: more than two players or two teams can participate in one game. Mease et al. [18] introduced a penalized maximal likelihood approach to rank NCAA college football teams. Their work also assumes that the intrinsic skill score of one team follows the normal distribution with fixed variance.

To summarize the relation with previous work, our approach (1) incorporates best answer selection to infer pairwise comparison of users, which includes pairwise comparisons between askers and best answerers, non-best answerers and best answerers, and leverages implicit question answering relationships between askers and answerers [14, 23]; (2) proposes a competition-based approach and then applies two-player competition models [10, 18, 6] to estimate the relative expertise score of users.

3. PAIRWISE COMPARISONS OF USERS

CQA is a virtual space where people can ask questions, seek opinions and get experiences from others. When an asker has a problem related to the topic of a certain category, he or she would ask a question within the certain category. Then, there will be several

answerers to answer his or her question. To guarantee the quality of content in CQA, the asker must select one answer as the best answer among all the answers he or she received within a fixed number of days after the question was posted. All participants in one question answering thread can be thought of as triplets (a, b, S) consisting of the asker a, the best answerer b whose answer is selected as the best answer, and the set S of all the other answerers who are named as non-best answerers. Figure 1(a) illustrates an example: the asker asked a question and got three answers from three answerers, and the answer posted by the third answerer (Answerer3) was selected as the best answer.



(a) Question answering (b) Pairwise competitions

Figure 1. Example for one question answering thread

Ideally, askers would not select the best answers randomly, but make an informed choice and the selected best answer should have the best quality among all answers. In reality, askers may be careless or subjective; their judgments may be not perfect. However, the best answer selections are still likely to convey some meaningful information. We make the following two intuitive assumptions about best answer selection:

1. Given a question, its best answerer b has a higher expertise level than its asker a. This is pretty straightforward since the best answerer successfully solves the problem that the asker doesn't know.

2. Given a question, its best answerer b has a higher expertise level than all other answerers, i.e. answerers in the set S. This is reasonable since the asker is expected to pick the best answers among all answers assuming the quality of an answer is positively correlated with the expertise level of its provider.

According to this intuition, there are $n = |S| + 1$ pairwise comparisons generated for the question answering thread with the asker a, the best answerer b and the non-best answerer set S. Taking a competition viewpoint, each pairwise comparison can be viewed as a two-player competition with one winner and one loser. Hence, there are n two-player competitions, including one competition between the asker a and the best answerer b, and $|S|$ competitions between the best answerer b and every non-best answerer in the set S. The best answerer b is the winner of each two-player competition, and all other users, including the asker a and all non-best answerers, are losers. Consider again the example from Figure 1. For the given question answering thread, there are three two-player competitions generated, including the one between Asker and Answerer3, the one between Answerer1 and Answerer3, and the one between Answerer2 and Answerer3. Answerer3 is the winner in these three two-player competitions, because his answer is selected as the best answer. Asker, Answerer1 and Answerer2 are all losers in those competitions. Hence, the problem of estimating the relative expert levels of users can be deduced to the problem of learning the relative skills of players from the win-loss results of generated two-player competitions.

Formally, the win-loss results of all two-player competitions generated from the thread q with the asker a, the best answerer b and non-best answerer set S can be represented as the following set:

$$R_q = \{(a \prec b), (s_1 \prec b), (s_2 \prec b), \cdots, (s_{|S|} \prec b)\}, \quad (1)$$

where $j \prec i$ means that user i bests user j.

Using $Q = \{q_1, \cdots, q_{n-1}, q_n\}$ to denote all questions in one category, the win-loss results of all two-player competitions generated from the set Q can be presented as the following set:

$$R = \{(j \prec i) \mid \forall (j \prec i) \in R_{q_i}, i = 1 \cdots |Q|\}. \quad (2)$$

Our problem is to learn the relative skills of players from the set R denoting all the win-loss results. In the next section, we present two competition-based models to solve this problem.

4. COMPETITION BASED MODELS

4.1 TrueSkill

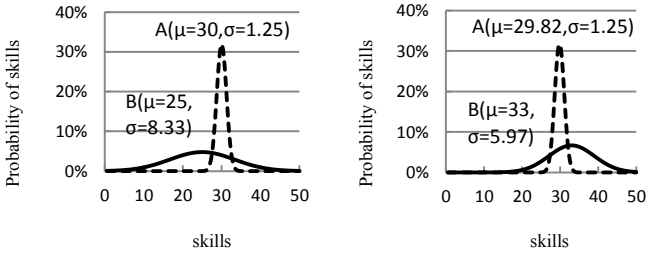
TrueSkill [10] is a Bayesian skill rating system which is designed to calculate the relative skill levels of players in multi-player or multi-player team games. As described in the previous section, our problem is to learn the relative expertise score of users from pairwise competitions without a tie. Hence, we introduce a two-player and no-draw version of TrueSkill to solve our problem.

TrueSkill assumes that the performance of each player in one game follows a normal distribution with its mean μ and its standard deviation σ. μ is the average skill of the player and σ represents a system's uncertainty about its estimation of the player's skill. Intuitively, as a system learns about the skill of one player from more data, the standard deviation σ (uncertainty) will be decreased. TrueSkill assumes that the skill of each player will be slightly changed after each game. This assumption both allows the system to track the skill improvement of players over time and guarantees that the standard deviation σ never decreases to zero. In the TrueSkill paper [10], $\mu - 3\sigma$ is used to rank players to ensure that the top ranked players are highly skilled with high certainty. In this paper, we follow the same approach to rank users.

Before going into details, we give a visual overview of what TrueSkill is. Figure 2 shows a simple example. There are two players: (a) A is an experienced player with a small standard deviation, since the estimation is based on many games and is therefore more certain; (b) B is a new player with a larger standard deviation since the system is not sure about B's skill. Figure 2 (a) shows the skill distributions of two players A and B before a game between them. Figure 2 (b) shows the updated skill distributions of two players after player B wins the new game. From Figure 2 (b), we can see that system makes a big update on the average skill μ of player B, because it considers that player B is probably better than player A based on the outcome of the new game. However, player B's standard deviation σ is still large, because the system is not confident about the estimation on B based on just one more game played by B.

In short, invoking Bayes' theorem, given the current estimated skills (priori probability) of players and the outcome of a new game (likelihood), a TrueSkill model should update its estimation

of player skills (posterior probability). Compared to our problem, the outcome of each game is from the set R defined in Equation (2). Taking the set *R,* whose elements are sorted by time, and setting initial value of average skill μ and standard deviation σ of each player, we can apply TrueSkill to estimate the relative expertise score of each user.



(a) Before the new game (b) After the new game

Figure 2. Example of updating player skill based on the outcome of a new game

The assumptions that updating average skill μ and standard deviation σ are intuitive: (a) the expected outcome is that the player with higher average skill wins the game, causing small updates on average skill μ and standard deviation σ; (b) the unexpected outcome is that the player with lower skill wins the game, causing large updates on average skill μ and standard deviation σ, to make the system more likely to predict the outcomes of future games. According to these assumptions, the equations to update the skills of players and the uncertainty about estimation are as follows:

$$\mu_{winner} = \mu_{winner} + \frac{\sigma_{winner}^2}{c} \cdot v\left(\frac{t}{c}, \frac{\varepsilon}{c}\right), \quad (3)$$

$$\mu_{loser} = \mu_{loser} - \frac{\sigma_{loser}^2}{c} \cdot v\left(\frac{t}{c}, \frac{\varepsilon}{c}\right), \quad (4)$$

$$\sigma_{winner}^2 = \sigma_{winner}^2 \cdot \left[1 - \frac{\sigma_{winner}^2}{c^2} \cdot w\left(\frac{t}{c}, \frac{\varepsilon}{c}\right)\right], \quad (5)$$

$$\sigma_{loser}^2 = \sigma_{loser}^2 \cdot \left[1 - \frac{\sigma_{loser}^2}{c^2} \cdot w\left(\frac{t}{c}, \frac{\varepsilon}{c}\right)\right], \quad (6)$$

where

$$t = \mu_{winner} - \mu_{loser},$$
$$c^2 = 2\beta^2 + \sigma_{winner}^2 + \sigma_{loser}^2,$$
$$v(t, \varepsilon) = \mathcal{N}(t - \varepsilon)/\Phi(t - \varepsilon),$$
$$w(\mathrm{t}, \varepsilon) = v(t, \varepsilon) \cdot (v(t, \varepsilon) + t - \varepsilon).$$

Here, $\mathcal{N}(\cdot)$ is the standard normal distribution, $\Phi(\cdot)$ is the cumulative normal distribution, ε is a parameter representing the probability of a draw in one game, and β is a parameter representing the range of skills. For example, the range of skills is large for a chess game, but it's small for gambling. In this paper, we set these two parameters to the value used in the TrueSkill paper [10]. The initial value of average skill μ and standard deviation σ of each player is also the same as the default value used in the TrueSkill paper [10].

The variable *t* reflects the exceptions on the outcome: (a) the outcome is expected, when *t* is positive; (b) the outcome is unexpected, when *t* is negative.

The function $v(t, \varepsilon)$ and $w(t, \varepsilon)$ are weighting factors to average skill μ and standard deviation σ, respectively. These two functions reflect the assumption about updating μ and σ. Figure 3 plots the tendency of function $v(t, \varepsilon)$ for a given ε. It can be observed that: (a) μ will not change too much when *t* is positive (expected result); (b) μ will change more when *t* is negative (unexpected result). Similarly, Figure 4 plots the tendency of function $w(t, \varepsilon)$ for a given ε. We can see that: (a) σ will not be changed too much when *t* is positive (expected result); (b) σ will be changed more when *t* is negative (unexpected result).

Besides the two weighting functions $v(t, \varepsilon)$ and $w(t, \varepsilon)$, another factor affecting the update on μ and σ is the ratio between the uncertainty of each player (σ_{winner} or σ_{loser}) and the total sum of uncertainties *c*. The player with a larger uncertainty gets a larger change on both μ and σ.

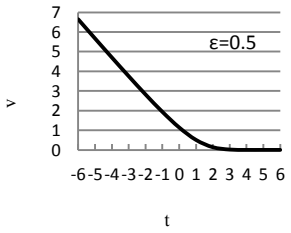


Figure 3. Example curve for function v

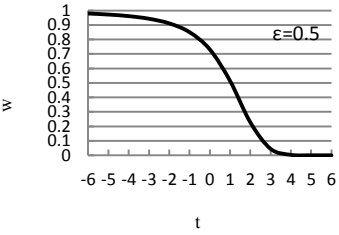


Figure 4. Example curve for function w

4.2 SVM Model

Mease et al. [18] proposed a maximal likelihood approach to rank football teams, which can be solved by a logistic regression model. Inspired by Mease's work, Carterette et al. [6] proposed an SVM model to solve the rank aggregation problem, which combines multiple search results from multiple search engines to produce a better new ranking. In the rank aggregation problem, for a given query, each search engine returns a ranked list of documents $d_1 \prec d_2 \prec d_3 \cdots$, where $d_j \prec d_i$ means that document d_i is ranked higher than document d_j. Each $d_j \prec d_i$ is viewed as a pairwise comparison between two documents (d_i, d_j). The SVM model learns the relevance weight of each document from these extracted pairwise comparisons.

In our problem, the pairwise comparisons of users $(j \prec i) \in R$ can be viewed as pairwise comparisons of documents $d_j \prec d_i$. Thus, we can apply the SVM model proposed by Carterette et al. [6] to learn the relative expertise score of users. The expertise score of each user *i* is defined as θ_i. In the SVM model, the optimization problem is:

$$\text{minimize } \frac{1}{2}\|\theta\|^2 + C\sum \xi_k \quad (7)$$
$$\text{subject to } y_k(\langle\theta, x_k\rangle + b) \geq 1 - \xi_k;\ \xi_k \geq 0$$

Where $\theta = \{\theta_1, \cdots, \theta_n\}$ and *n* is the number of all users. Here x_k is a vector of length *n* associated with a pairwise competition between users, and *y* is the win-loss result. Given a two-player competition *k* with the winner *i* and the loser *j*, there're two training instances generated: (a) $y_k = 1$, $x_k[i] = 1$, $x_k[j] = -1$; (b) $y_k' = 0$, $x_k'[i] = -1$, $x_k'[j] = 1$. The $\hat{\theta} = (\hat{\theta}_1, \cdots, \hat{\theta}_n)$ that minimize equation (7) are taken as the estimated relative skills of players. In this paper, we use linear kernel SVM LIBLINEAR [9] to solve this optimization problem.

5. EXPERIMENTS

5.1 Data Set

In this paper, we use the NTCIR-8 CQA task data as the experiment data, which is dumped from the Yahoo! Chiebukuro (Japanese Yahoo! Answers) database ranging from April 2004 to October 2005. We choose this corpus because it is the only publically available CQA data with multiple manual answer quality judgments. It contains 3,116,009 questions, 13,477,785 answers including 3,116,008 best answers (one best answer is missed from the data), and 240,784 users. There are 14 categories provided in the data set. Each question belongs to exactly one category. Figure 5 shows the frequency distribution of questions and the frequency distribution of users over the 14 different categories. The NTCIR-8 CQA task organizers sampled 1,500 questions from the entire data set according to the frequency distribution of questions over 14 categories as the test data set. There are 7,443 answers and 6,482 users in the test data set. It includes 1,500 best answers which we denote as BA data. In the testing data, the number of answers per question ranges from 2 to 20.

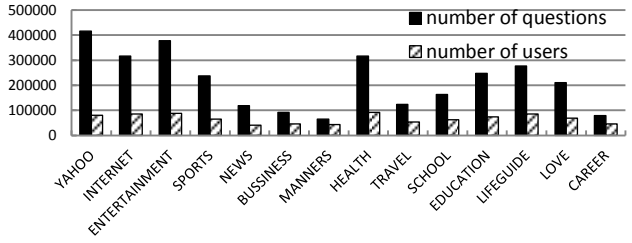


Figure 5. The frequency distribution of questions and users over the 14 top categories

The NTCIR-8 CQA task was originally designed for evaluating answer quality prediction systems. Sakai et al. [20, 21] state that the best answers selected by askers may be biased, and that there may be other good answers besides the best answers. To solve these two problems, NTCIR-8 CQA task organizers hired four assessors to annotate answer quality and assign a graded-relevance score to each answer using a pyramid approach. There are 9 relevance levels from L0 (low) to L8 (high) defined in the testing data. Table 1 shows the number of answers for each relevance level. We shall refer to this ground truth data set as graded answers (GA) data. Table 2 shows the relationship between BA data and GA data. We observe that not all BA are high quality answers based on GA and that there are other good answers besides BA. It also shows that overall BA is still a good answer quality indicator since the number of good BA according to GA is increasing at each higher GA relevance level.

Table 1. Number of answers at each relevance level (GA)

L0	L1	L2	L3	L4	L5	L6	L7	L8	Total
17	32	106	238	1318	1399	1527	1505	1301	7443

Table 2. Number of BAs at each GA relevance level

L0	L1	L2	L3	L4	L5	L6	L7	L8	Total
0	1	7	22	99	156	245	372	598	1500

5.2 Evaluation Metrics

There are two major approaches for the evaluation of user expertise estimation or expert identification: (1) employing traditional information retrieval (IR) evaluation metrics, such as precision, recall and rank correlation etc., to measure system output using a ground truth (an expert set or a user ranked list); (2) evaluating the quality of answers posted by identified experts.

How to get the ground truth is an important problem for the first evaluation approach. Two types of ground truth were used in previous work: (1) an automatically generated user ranked list; (b) a manually annotated expert set. However, both ground truths are not perfect. Jurczyk et al. [14] use several meta-data information sets in CQA to generate a user ranked list as the ground truth, including the average number of votes received by one user, the average number of stars attained by one user, and the best answer ratio of one user. Bian et al. [2] used the top contributor list provided by Yahoo! Answers as the ground truth. The top contributor badge[3] in Yahoo! Answers is automatically computed according to the recent number of best answers and best answer ratio of users. Unfortunately, such an automatic generated ground truth is obtained according to a certain heuristic method, which itself can be viewed as an approach to estimate user expertise level. Thus, the automatic generated ground truth may be not accurate. Pal et al. [18] used an expert set that is annotated by the employees of a CQA site TurboTax as the ground truth. Zhang et al. [23] asked two domain experts to assign an expertise level for each frequent user in an online community according to the posting history of users. However, it's hard to track even a small amount of users in a large online community. The manually annotated expert set may be not up to date or cannot cover all experts in a community.

As discussed previously, Bouguessa et al. [5] consider the problem of how many users should be selected as experts from a user ranking list that is sorted by number of best answers. They proposed an indirect evaluation metric to evaluate their method. They used an automatic answer quality prediction system to evaluate the average quality of answers posted by experts identified by their method. Their assumption is simply that experts are expected to generate high-quality answers. However, automatic answer quality prediction may not be accurate enough.

In this paper, we consider how to estimate the relative expertise scores of users. Therefore, we are interested in evaluating the relative rank of users sorted by their estimated user expertise score. Inspired by Bouguessa's work [3], we assume that the higher the expertise level of a user, the higher the quality of answer provided by the user. Given a testing question, its answers can be sorted by estimated user expertise scores of answerers. Hence, evaluating answer ranking can be a validation of evaluating the relative ranking of users. Additionally, we use a human annotated data set, including GA data and BA data, to evaluate system output, rather than using an automatic answer quality prediction system.

Using BA data, we treat the best answer as the only right answer and evaluate answer ranking with two metrics –Mean Reciprocal Rank (MRR) and Precision@1 (P@1). However, using BA data, we only can do binary judgments. As described previously, there are many other good answers besides the best answers. It's therefore better to use GA data to evaluate. Hence, GA data will be used as the main ground truth.

For GA data, we use two graded-relevance evaluation metrics: nDCG (normalized Discounted Cumulative Gain) and RnDCG (Relatively normalized Discounted Cumulative Gain). In the GA

[3] http://help.yahoo.com/l/us/yahoo/answers/network/contributor.html

data, let the gain values be 0-8 for L0-L8 relevant answers respectively. Let $g(r)$ denote the gain value of the answer ranked at r in a system's output. Similarly, let $g^*(r)$ denote the gain value of the answer ranked at r in the best possible ranking list obtained by sorting all answers in non-ascending order of the gain values. The nDCG score with cutoff n is defined as:

$$\mathrm{nDCG@}n = \frac{\mathrm{systemDCG@}n}{\mathrm{bestDCG@}n}, \quad (8)$$

where

$$\mathrm{systemDCG@}n = \sum_{\mathrm{r}=1}^{\mathrm{n}} \frac{\mathrm{g(r)}}{\log(\mathrm{r}+1)},$$

$$\mathrm{bestDCG@n} = \sum_{\mathrm{r}=1}^{\mathrm{n}} \frac{\mathrm{g^*(r)}}{\log(\mathrm{r}+1)}.$$

As shown in Table 1, there are 94.7% answers that are equal to or greater than level L4 and that only 0.23% of answers are at level L0 (totally irrelevant). It means that even a bad system can get a high nDCG score. It's different from the standard IR evaluation, because there are much more irrelevant documents in many other IR tasks. Hence, in our particular case, we use a relative normalization approach (RnDCG) to normalize the DCG score (suggested in Sakai et al. [21]), which ensures that the evaluation scores will range fully between 0 and 1 for better differentiating between systems. Let $g^{\Delta}(r)$ denote the gain value of the answer ranked at r in the worst possible ranking list obtained by sorting all answers in non-descending order of the gain values. The RnDCG score with cutoff n can be defined as follows:

$$\mathrm{RnDCG@}n = \frac{\mathrm{systemDCG@}n - \mathrm{worstDCG@}n}{\mathrm{bestDCG@}n - \mathrm{worstDCG@}n}, \quad (9)$$

where

$$\mathrm{worstDCG@}n = \sum_{r=1}^{n} \frac{g^{\Delta}(r)}{\log(r+1)}.$$

In our experiment, we use the entire NTCIR CQA data set (excluding the 1,500 testing questions) as training data to learn the relative expertise score of users for each category. For each testing question within a certain category, all the answers to the question are sorted by the estimated expertise score of their authors. Then, the evaluation metrics for answer ranking can be applied to measure the performances of the user expertise score estimation approaches. In this paper, RnDCG on GA data will be used as the main evaluation metric.

5.3 Baseline Methods

Table 3 lists the user expertise score estimation methods which are evaluated in this paper and their abbreviations. The number of answers and the number of best answers were used as the simplest baselines in [14, 3, 23]. Link analysis approaches can be applied on a question answering relationship based user graph (QA based user graph). In the QA based user graph, there is a directed edge from one asker to its answerer. PageRank and HITS have been applied on the QA based user graph [23, 15]. The QA based user graph assumes that all answers have equal quality. However, the quality of different answers varies drastically in CQA sites [1] as we have observed even at the best answer level in Table 2. Hence, we simply propose to build a user graph based on the question and best answering relationship (QBA based user graph). In the QBA based user graph, there is a directed edge from an asker to its best answerer. The methods running PageRank and HITS on the QBA based user graph are used as the other two baselines.

Table 3. The methods and their abbreviations

Method	Abbrev.
Number of Answers (Sec. 2)	NA
Number of Best Answers (Sec. 2)	NBA
PageRank on QA based user Graph(Sec. 2)	P+QAG
PageRank on QBA based user Graph	P+QBAG
HITS on QA based user Graph(Sec. 2)	H+QAG
HITS on QBA based user Graph	H+QBAG
Best Answer Ratio	BAR
Smoothed Best Answer Ratio	SBAR
TrueSkill (Sec. 4.1)	TS
SVM Model (Sec. 4.2)	SVM

Another simple method we used as a baseline is the best answer ratio (BAR). The BAR of one user u can be computed as follows:

$$BAR(u) = \frac{C(BA;u)}{C(A;u)}, \quad (10)$$

where $C(A;u)$ denotes the number of answers provided by user u, and $C(BA;u)$ denotes the number of best answers provided by user u. To the best of our knowledge, there's no related work using the BAR as a user expertise score estimation method and comparing it with other methods[4]. It was only used as an effective feature to predict answer quality [1, 12, 17].

The BAR might be overestimated or underestimated when $C(A;u)$ is small. For example, given two users A and B, A only posts 1 answer and gets 1 best answer; while B posts 100 answers and has 90 best answers. In this case, A's BAR is higher than B's but we really are not sure that A is really better than B due to the low count of A's answers. In another case, the BAR for a user posting 1 answer and having no best answer is zero, which may be lower than his or her true expertise level. Hence, we propose a smoothed best answer ratio (SBAR) method which considers the number of answers given by a user. It is computed as follows:

$$SBAR(u) = \frac{C(A;u)}{C(A;u)+\alpha} BAR(u) + \frac{\alpha}{C(A;u)+\alpha} BAR_{avg}, \quad (11)$$

where

$$\alpha = \frac{1}{|U|} \cdot \sum_{\forall v \in U} C(A;v), \qquad BAR_{avg} = \frac{1}{|U|} \cdot \sum_{\forall v \in U} BAR(v).$$

Here, $|U|$ denotes the total number of users, α means the average number of answers per user, and BAR_{avg} means the average BAR per user. From Equation (11) we see: if the number of answers posted by a user is small (less than the average number of answers per user), his or her score will be smoothed toward the average score of all users; otherwise it will be close to the maximum likelihood estimation of its BAR.

[4] As discussed in section 5.2, the BAR was used as a method to get the ground truth of user ranking in Jurczyk et al. [14].

Table 4. The performance of all methods on BA and GA data for two user sets.

		GA data						BA data	
	Method	RnDCG@1	RnDCG@3	RnDCG@20	nDCG@1	nDCG@3	nDCG@20	P@1	MRR
Test users with at least 50 answers, 975 selected questions	NA	0.4874	0.5090	0.5035	0.8044	0.9033	0.9455	0.2954	0.5567
	NBA	0.5462	0.5658	0.5577	0.8273	0.9160	0.9516	0.3426	0.5876
	P+QAG	0.5043	0.5184	0.5158	0.8107	0.9050	0.9468	0.3015	0.5604
	P+QBAG	0.5462	0.5631	0.5588	0.8267	0.9153	0.9517	0.3528	0.5964
	H+QAG	0.4866	0.5073	0.5024	0.8044	0.9033	0.9454	0.2974	0.5576
	H+QBAG	0.5424	0.5645	0.5565	0.8268	0.9161	0.9517	0.3426	0.5886
	BAR	0.6684	0.6875	0.6802	0.8767	0.9420	0.9660	0.4349	0.6494
	SBAR	0.6687	0.6885	0.6808	0.8770	0.9425	0.9661	0.4390	0.6519
	TS	0.6738	0.7011	0.6899	0.8779	0.9426	0.9666	0.4349	0.6505
	SVM	**0.6939**	**0.7145**	**0.7061**	**0.8871**	**0.9467**	**0.9688**	**0.4523**	**0.6637**
Test users with at least 1 answer, 1463 selected questions	NA	0.4733	0.4932	0.4905	0.7901	0.8862	0.9409	0.2632	0.5160
	NBA	0.5197	0.5385	0.5338	0.8087	0.8976	0.9460	0.2960	0.5385
	P+QAG	0.4865	0.5012	0.5010	0.7945	0.8872	0.9418	0.2700	0.5193
	P+QBAG	0.5170	0.5395	0.5341	0.8068	0.8972	0.9458	0.3035	0.5457
	H+QAG	0.4738	0.4951	0.4923	0.7903	0.8865	0.9411	0.2659	0.5174
	H+QBAG	0.5175	0.5410	0.5355	0.8086	0.8981	0.9464	0.2973	0.5407
	BAR	0.6534	0.6700	0.6636	0.8659	0.9291	0.9622	**0.3999**	0.6134
	SBAR	0.6585	**0.6812**	**0.6728**	**0.8676**	**0.9316**	**0.9633**	**0.3999**	**0.6155**
	TS	0.6486	0.6636	0.6568	0.8631	0.9272	0.9612	0.3821	0.6018
	SVM	**0.6587**	0.6760	0.6688	**0.8676**	0.9314	0.9629	0.3841	0.6092

5.4 Results

As shown in Table 4, we conduct evaluations on two user sets: (a) the set of all users who posted at least 50 answers in the training data; (b) the set of all users who posted at least 1 answer in the training data.

To test on the first user set, we select a subset of questions from the 1500 testing questions with these two properties: (1) answered by at least two users from the first user set; (2) their best answerers are also from the first user set. It should be noticed that there can be answerers who are not from the first user set (unseen users) and participated the selected testing questions. Hence, in this selected testing question set, the answers posted by unseen users will be removed and only the answers posted by the users from the first user set will be kept. It ensures that only the answers posted by users from the first user set will be evaluated. Then answer quality prediction based evaluation metrics can be applied. There are 975 questions in this selected question set. Similarly, we create another set of testing questions for the second user set. There are 1463 questions in the second selected question set.

The purpose of conducting the first evaluation is to measure the performances of different methods on the set of active users who post many answers. This is because the active users contribute a lot to communities and are the driving force of communities. In our data set, there are 12.5% users who provide more than 50 answers and contribute more than 91.6% answers. Therefore, it would be very beneficial to site owners or CQA researchers to learn more about active users and differentiate between their relative expertise levels. In contrast, the goal of the second evaluation is to measure the performance of different methods on the set of all answerers including the users posting only a few answers. This is because new users that post a small number of answers are the potential new driving force of communities. If a method can well estimate the expertise scores of new users with just a few answers, it would be highly beneficial for online communities.

5.4.1 Answer vs. Best Answer

In this section, we show the effect of using the best answer versus using answer on counting based methods, i.e. NA vs. NBA, and on link analysis based methods, i.e. P+QAG vs. P+QBAG, and H+QAG vs. H+QBAG.

As shown in Table 4, comparing NA with NBA, P+QAG with P+QBAG, and H+QAG with H+QBAG, we found that by incorporating the best answer, the performances of the counting and link analysis based methods are significantly improved in terms of all evaluation metrics on two user sets (Wilcoxon signed-rank test, p-value < 0.01). It partially proves that answer quality (best answer) is important for user expertise score estimation.

As shown in Table 4, P+QAG is slightly better than NA in terms of all evaluation metrics on two different user sets. H+QAG is slightly better than NA in terms of all evaluation metrics on the second user set. However, sometimes it's worse than NA on the first user set. It's similar to what Jurczyk et al. [14] reported: that HITS doesn't perform well sometimes. It's also reported by Zhang et al. [23] that sometimes a relatively simple measure is as good as a complex algorithm such as PageRank. Comparing NBA with P+QBAG and H+QBAG, we come to a similar conclusion that link analysis (or graph) based approaches perform similar to the counting based methods (NA, NBA).

5.4.2 Best Answer Ratio

As we described in section 5.3, to the best of our knowledge, there's no related work comparing such simple, intuitive and strong methods BAR and SBAR with other user expertise estimation methods. As Table 4 shows, it's surprising that simple BAR and SBAR can significantly outperform more complex methods, such as P+QBAG and H+QBAG, in terms of all evaluation metrics on two user sets (Wilcoxon signed-rank test, p-value < 0.01).

Also, we can observe that SBAR is slightly better than BAR in terms of all evaluation metrics on two user sets, since SBAR incorporates smoothing into BAR to avoid over fitting.

Figure 6 shows the distributions of SBAR scores of all answerers in two categories: internet and travel. We observe that the trends of the two distributions are similar (similar trends can be observed in other categories). From Figure 6, we see that the curve denoting the distribution of answerers' SBAR scores can be roughly divided into three parts: short head, long middle, and short tail. The users in the short and sharp head part are the ones with high SBAR scores. The users in the short tail part are the ones with low SBAR scores. One interesting finding is that most of users that fall into the long and flat middle are low frequent users with negative participation patterns. The negative user participation patterns are like the ones highlighted in Figure 6: the negative participation patterns are one answer with zero best answers (1:0), two answers with one best answer (2:1) and two answers with zero best answers (2:0). This reflects Yang et al.'s [13] finding: if an answerer didn't get positive feedback, i.e. selected as the best answer at the initial participation, it is very likely the answerer would stop contributing to a community. Intuitively, a user's BAR score should be highly correlated with continued answering. Because it's likely that only the users who always get positive feedbacks from a community, i.e. selected as the best answer, would like to continuously contribute to the community. However, Yang et al.'s [13] reported that a user's BAR score is just weakly correlated with continued answering. Figure 7 shows the number of answers by users from each bin of SBAR scores. The tail part in Figure 7 tells us the reason is that there were a lot of users who continuously tried to answer questions, even if they always failed.

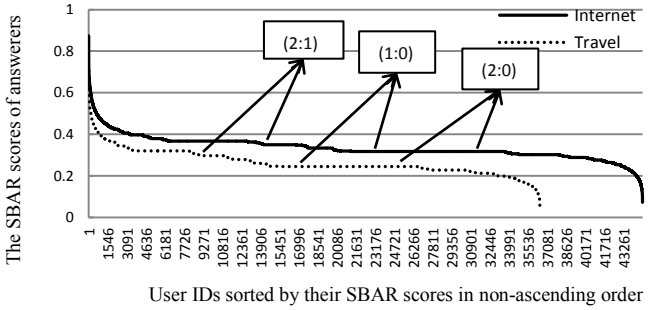


Figure 6. The distribution of answerers' SBAR scores

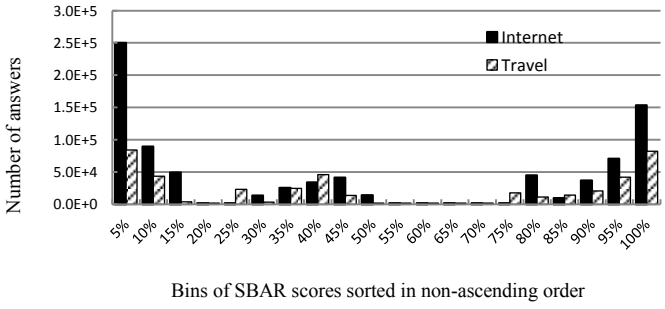


Figure 7. Number of answers for each bin of SBAR scores

From these observations, we find that SBAR can easily distinguish between two user groups: the short head part and the short tail part. However, it doesn't consider interactions between users. If we can take advantage of frequent interactions between high SBAR users and apply competition-based models to estimate relative user expertise score, we might get even better estimation results than using SBAR alone. We show that this can be achieved in the next section.

5.4.3 Best Answer Ratio vs. Competition-based Methods

From the experiment results on the first user set in Table 4, we can see that SVM significantly outperforms TS in terms of all evaluation metrics using GA data (Wilcoxon signed-rank test, in terms of RnDCG@1, RnDCG@3, nDCG@1 and nDCG@3 p-value<0.05; in terms of RnDCG@20 and nDCG@20, p-value<0.01). From the experimental results on the second user set in Table 4, we can see that SVM also outperforms TS in terms of the evaluation metrics using GA data (Wilcoxon signed-rank test, in terms of RnDCG@3, RnDCG@20, nDCG@3 and nDCG@20, p-value<0.01). The reason that SVM can outperform TS is that SVM considers the all pairwise comparisons globally; while TS considers each pairwise comparison one by one. TS was originally designed to track the changes of user skill by assuming that a user's skill will change over time. In our case, a user's expertise level will not change too much within a short period of time.

Comparing BAR and SBAR with TS and SVM on the first user set, we see that SVM significantly outperforms BAR and SBAR in terms of all evaluation metrics using GA data (Wilcoxon signed-rank test, in terms of RnDCG@1 and nDCG@1, p-value<0.05; in terms of RnDCG@3, RnDCG@20, nDCG@3 and nDCG@20, p-value<0.01). Also, it can be observed that TS outperforms BAR and SBAR in terms of all evaluation metrics using GA data (though, it doesn't pass the significant test). It shows that estimating relative user expertise scores of active users can benefit from modeling the interactions between users.

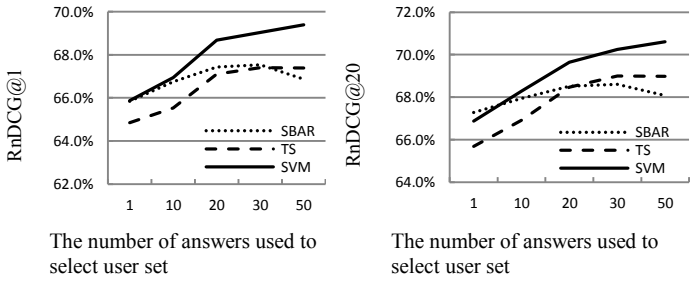


Figure 8. The performance of SBAR, TS and SVM on different user sets, in terms of RnDCG@1 and RnDCG@20

However, it can be observed that SBAR outperforms TS and SVM on the second user set. The reason is that competition-based models (TS and SVM) usually need more data about pairwise comparisons between users to learn well. However, for the low frequent users, there's only a small amount of interaction between them, which may be not enough for a pairwise comparison based competition model to learn the relative expertise scores well. If this is true, it would be interesting to know how active a user needs to be so that competition-based models (TS and SVM) can perform reasonably well and better than the strong baseline SBAR. Hence, we evaluate SBAR, TS and SVM on 5 different user sets selected by different active levels measured by the number of answers posted per user. Figure 8 shows the performance of SBAR, TS and SVM on 5 different user sets, in terms of RnDCG@1 and RnDCG@20, respectively. SVM performs much better than SBAR for the users who posted more than 20 answers. The trend of the TS performance curve is similar. With the increment of the number of answers, the performances of TS and SVM become better.

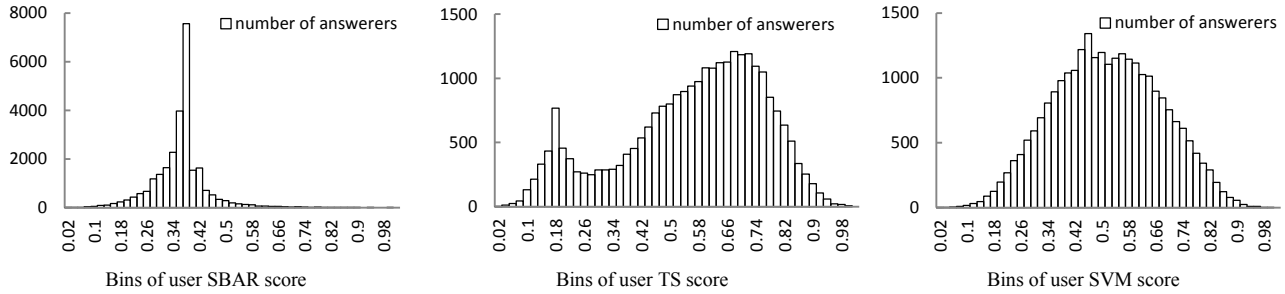


Figure 9. The frequency distributions of user score estimated by three methods (SBAR, TS and SVM) on internet category; only the answerers with at least one best answer are considered, user expertise score has been normalized into [0,1]

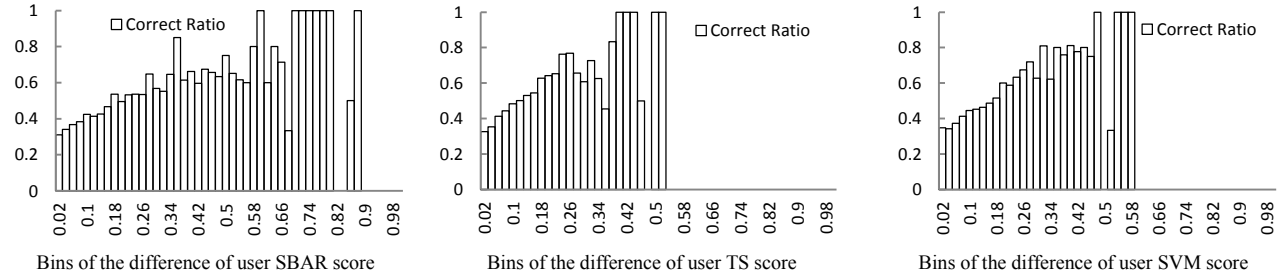


Figure 10. The ratio of correctly predicting pairwise competitions distributed over the difference of two users' expertise estimated by three methods (SBAR, TS and SVM); user expertise score has been normalized into [0,1]

In summary, the experimental results show that pairwise comparison based competition models (TS and SVM) perform better on active user sets. Specifically, SVM significantly outperforms two strong baselines, BAR and SBAR, on active user sets.

Figure 9 shows the frequency distributions of user expertise score estimated by three methods (SBAR, TS and SVM) in the internet category (it's similar in other categories). We make a very interesting observation that the distribution of the SVM score follows a normal distribution, which somewhat reflects our understanding of real world user behavior: (a) the expertise levels of most users are around the average level; (b) there is a small number of users with high expertise levels; (c) also, there is a small number of users with low expertise levels; this is probably due to the fact that users of low expertise are less likely to participate in knowledge sharing communities rather than lack of them. Comparing the score distributions of SBAR, TS, and SVM, we see that the distribution of SBAR score is much sharper.

5.5 Discriminative Power

Given a testing question with *n* answerers, there are $n(n-1)/2$ pairwise competitions between each two users that can be generated. Let $exp(u)$ denote the expertise score of user *u* estimated by a given method. Let $rel(a_u)$ denote the relevance level of the answer *a* provided by user *u*. For a pairwise competition (u_i, u_j), we say that there are three possible outcomes; (1) u_i is the winner and u_j is the loser when $rel(a_{u_i})$ is larger than $rel(a_{u_j})$; (2) u_i is the loser and u_j is the winner when $rel(a_{u_i})$ is smaller than $rel(a_{u_j})$; (3) it's a tie when $rel(a_{u_i})$ is equal to $rel(a_{u_j})$. We can use the sign of difference between $exp(u_i)$ and $exp(u_j)$ to predict the outcome of the given pairwise competition.

We say the prediction is correct when

$$sgn(exp(u_i) - exp(u_j)) = sgn(rel(a_{u_i}) - rel(a_{u_j})) \quad (12)$$

where

$$sgn(x) = \begin{cases} 1 & if\ x > 0 \\ 0 & if\ x = 0 \\ -1 & if\ x < 0 \end{cases}$$

In this section, we use the first selected question set in Table 4 as the testing data. There are 9,417 pairwise competitions generated. Table 5 shows the performance of SBAR, TS and SVM in predicting the outcome of the generated pairwise competitions. We see that SVM and TS perform better than SBAR.

Table 5. The performance of three methods (SBAR, TS and SVM) to predict the outcome of pairwise competitions

	Incorrect Pairs	Correct Pairs	Error Rate
SBAR	5272	4145	55.98%
TS	5269	4148	55.95%
SVM	**5161**	**4256**	**54.81%**

Furthermore, given a pairwise competition, it's expected that the larger the difference of two users' estimated expertise score, the higher the probability of correct prediction. In contrast to this, it's expected that the smaller the difference, the lower the probability of correct prediction. When the difference is small, it means that the given user expertise score estimation method cannot differentiate between two users well. Discriminative power is defined as the averaged ratio of correct predication at a given difference. Figure 10 shows the discriminative power of three methods (SBAR, TS and SVM) distributed over the difference of two users' expertise scores. Basically, we see that it holds for all three methods that the larger the difference, the higher the ratio of correct prediction. Additionally, we find that when the difference is smaller than a certain number, the ratio of correct predictions will be less than 0.5 (random). If this case happened when we applying user expertise scores on some downstream applications, we could not trust the prediction result by the user expertise score. Hence, the knowledge of the discriminative power of a score method can provide an informed guideline for other downstream applications

on how to utilize the estimated user expertise score. For example, only use expertise score information when the score difference is within the discriminative power of the scoring method.

Because the scales of user expertise score output by different methods are different, it's hard to compare the discriminative power of different methods directly. Inspired by R. Herbrich et al. [10], we set six challenge competitions for SBAR, TS, and SVM. Let each method consider which pairwise competitions in the testing data are the most difficult to predict and present them to other methods. Using one user expertise score estimation method (defender), we can sort all 9,417 pairwise competitions in the testing data in non-descending order of the difference between two users' expertise scores and present the top 2,000 pairwise competitions for another user expertise score estimation method (challenger) to predict results. Table 6 shows the success ratio (correct prediction) of each challenger in each challenge competition between two user expertise score estimation methods. We see that SVM beats both TS and SBAR, and TS beats SBAR. It means that the pairwise competition based user expertise estimation method has better discriminative power than SBAR.

Table 6. Comparing the discriminative power of three methods (SBAR, TS and SVM) by using challenging modes

		Challenger		
		SBAR	TS	SVM
Defender	SBAR	N/A	694(34.7%)	**729(36.5%)**
	TS	679(34.0%)	N/A	**724 (36.2%)**
	SVM	**664(33.2%)**	**661 (33.1%)**	N/A

6. CONCLUSTION AND FEATURE WORK

In this paper, we presented two competition-based methods, TS and SVM, to estimate the relative expertise scores of users in CQA. We proposed two simple and intuitive principles and leveraged best answer selection to cast the relative expertise score estimation problem as a problem of relative skill estimation in two-player games where competition-based methods such as TS and SVM can be readily applied to estimate user expertise scores. For evaluation, we introduced an answer quality prediction based evaluation metric and used NTCIR-8 CQA data. We also are the first to introduce the idea of the relation between the discriminative power of scoring methods and their expected performance. Experimental results show that: (1) competition-based models significantly outperform link analysis based methods and pointwise methods; (2) competition-based models have better discriminative power. We also found that competition-based methods perform better when they have enough active users.

Future work may follow two paths: (1) expand the competition-based method into forums by incorporating automatic answer quality prediction; (2) analyze user knowledge to help find subject experts.

7. ACKNOWLEDGEMENT

We would like to thank Tao Qin and Yunbo Cao for their valuable suggestions on this paper, Matt Callcut for his proofreading of this paper, and the anonymous reviewers for their helpful comments on this work.

8. REFERENCES

[1] E. Agichtein, C. Castillo, D. Donato, A. Gionis, and G. Mishne. Finding high-quality content in social media. In *Proc. WSDM*, pages 183–194. 2008.

[2] J. Bian, Y. Liu, D. Zhou, E. Agichtein, and H. Zha. Learning to recognize reliable users and content in social media with coupled mutual reinforcement. In *Proc. WWW*, pages 51–60. 2009.

[3] M. Bouguessa, B. Dumoulin, and S. Wang. Identifying authoritative actors in question-answering forum: the case of Yahoo! answers. In *Proc. SIGKDD*, pages 866–874. 2008.

[4] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. In *Proc. WWW*. 1998.

[5] C. Campbell, P. Maglio, A. Cozzi, and B. Dom. Expertise identification using email communications. In *Proc. CIKM*, pages 528–531. 2003.

[6] B. Carterette and D. Petkova. Learning a ranking from pairwise preferences. In *Proc. SIGIR*, pages 629–630. 2006.

[7] G. Cong, L. Wang, C.-Y. Lin, Y. Song, and Y. Sun. Finding question-answer pairs from online forums. In *Proc. SIGIR*, pages 467–474. 2008.

[8] Elo. *The rating of chessplayers, past and present*. Batsford, 1978.

[9] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874, 2008.

[10] R. Herbrich, T. Minka, and T. Graepel. TrueSkill: A Bayesian skill rating system. In *Proc. NIPS*, 20:569–576, 2007.

[11] D. Horowitz and S. Kamvar. The anatomy of a large-scale social search engine. In *Proc. WWW*, pages 431–440. 2010.

[12] J. Jeon, W. Croft, J. Lee, and S. Park. A framework to predict the quality of answers with non-textual features. In *Proc. SIGIR*, pages 228–235. 2006.

[13] Y. Jiang, W. Xiao, M. Ackerman, and L. Adamic. Activity lifespan: An analysis of user survival patterns in online knowledge sharing communities. In *Proc. AAAI*, 2010.

[14] P. Jurczyk and E. Agichtein. Discovering authorities in question answer communities by using link analysis. In *Proc. CIKM*, pages 919–922. 2007.

[15] J. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM (JACM)*, 46(5):604–632, 1999.

[16] B. Li and I. King. Routing Questions to Appropriate Answerers in Community Question Answering Services. In *Proc. CIKM*. 2010.

[17] Y. Liu, J. Bian, and E. Agichtein. Predicting information seeker satisfaction in community question answering. In *Proc. SIGIR*, pages 483–490. 2008.

[18] D. Mease. A penalized maximum likelihood approach for the ranking of college football teams independent of victory margins. *The American Statistician*, 57(4):241–248, 2003.

[19] Pal and J. Konstan. Expert identification in community question answering: exploring question selection bias. In *Proc. CIKM*. 2010.

[20] T. Sakai, D. Ishikawa, and N. Kando. Overview of the NTCIR-8 Community QA Pilot Task (Part II): System Evaluation. In *Proc. NTCIR-8*, pages 433–457, 2010.

[21] T. Sakai, D. Ishikawa, N. Kando, Y. Seki, K. Kuriyama, and C.-Y. Lin. Using Graded-Relevance Metrics for Evaluating Community QA Answer Selection. In *Proc. WSDM*. 2011.

[22] M. Suryanto, E. Lim, A. Sun, and R. Chiang. Quality-aware collaborative question answering: methods and evaluation. In *Proc. WSDM*, pages 142–151. 2009.

[23] J. Zhang, M. Ackerman, and L. Adamic. Expertise networks in online communities: structure and algorithms. In *Proc. WWW*, pages 221–230. 2007.

[24] D. Zhou, S. Orshanskiy, H. Zha, and C. Giles. Co-ranking authors and documents in a heterogeneous network. In *Proc. ICDM*, pages 739–744. 2008.