NCR: A Scalable Network-Based Approach to Co-Ranking in Question-and-Answer Sites

Jingyuan Zhang
University of Illinois at Chicago
Chicago, IL, USA
jzhan8@uic.edu

Xiangnan Kong
University of Illinois at Chicago
Chicago, IL, USA
xkong4@uic.edu

Luo Jie
Yahoo! Labs
Sunnyvale, CA, USA
luojie@yahoo-inc.com

Yi Chang
Yahoo! Labs
Sunnyvale, CA, USA
yichang@yahoo-inc.com

Philip S. Yu
University of Illinois at Chicago
Chicago, IL, USA
psyu@uic.edu

ABSTRACT

Question-and-answer (Q&A) websites, such as Yahoo! Answers, Stack Overflow and Quora, have become a popular and powerful platform for Web users to share knowledge on a wide range of subjects. This has led to a rapidly growing volume of information and the consequent challenge of readily identifying high quality objects (questions, answers and users) in Q&A sites. Exploring the interdependent relationships among different types of objects can help find high quality objects in Q&A sites more accurately. In this paper, we specifically focus on the ranking problem of co-ranking questions, answers and users in a Q&A website. By studying the tightly connected relationships between Q&A objects, we can gain useful insights toward solving the co-ranking problem. However, co-ranking multiple objects in Q&A sites is a challenging task: a) With the large volumes of data in Q&A sites, it is important to design a model that can scale well; b) The large-scale Q&A data makes extracting supervised information very expensive. In order to address these issues, we propose an unsupervised **N**etwork-based **C**o-**R**anking framework (NCR) to rank multiple objects in Q&A sites. Empirical studies on real-world Yahoo! Answers datasets demonstrate the effectiveness and the efficiency of the proposed NCR method.

Categories and Subject Descriptors

H.2.8 [**Database management**]: Database applications-Data mining

Keywords

Q&A Networks; Unsupervise; Co-Ranking; Interrelationships

CIKM'14, November 3–7, 2014, Shanghai, China.
Copyright 2014 ACM 978-1-4503-2598-1/14/11 ...$15.00.
http://dx.doi.org/10.1145/2661829.2661978 .

1. INTRODUCTION

Over the past few years, question-and-answer (Q&A) sites, such as Yahoo! Answers, Stack Overflow and Quora, have provided a new way for Web users to share knowledge on a wide range of subjects. Individuals can conveniently address specific needs to the public and get first-hand replies. These Q&A sites have exploded in popularity. Yahoo! Answers, the first and largest Q&A site, had 7,000 new questions and 21,000 new answers posted per hour in July, 2012 [18]; Stack Overflow, a Q&A site for computer programmers, had over 1.9 million registered users and more than 5.5 million questions in August, 2013. These repositories of valuable knowledge provide a gold mine for information retrieval and automatic question answering.

Since the Q&A site allows anyone to contribute knowledge on the Web, the quality of its objects (questions, answers and users) varies dramatically. On Yahoo! Answers, the answers provided by experts are detailed and useful, while others, provided by non-experts, may even contain spam and junk information. This is true for other types of objects as well. Some questions become popular in a short period of time and they get thousands of answers. In contrast, some questions are ridiculous and fail to get any answers. Distinguishing high quality objects from low quality ones can help improve the service offered by Q&A websites. Therefore, the problem of object ranking in Q&A sites has received considerable attention in the last few years [13, 10, 29].

Most conventional approaches of object ranking in Q&A sites focus on a single type of object. For instance, the works in [2, 13] investigated how to evaluate or predict the quality of questions, and [29, 23] aim to find high quality answers. In reality, the multiple types of objects are interrelated, e.g., good questions often attract high quality answers from competent users. Exploring the interrelationships among different types of objects can help identify high quality objects in Q&A sites more effectively.

In this paper, we specifically focus on the ranking problem of co-ranking questions, answers and users in a Q&A website. Studying the co-ranking problem has many benefits to real-world applications. With high quality objects, Q&A websites can recommend the most trendy information to users. In addition, the top objects under a certain topic or over all the topics can help improve user engagements because individuals can have a quick tour over popular information in the community. Furthermore, the co-ranking performance can be considered as an important signal for answer vertical search and web search. Integrated with such query independent signal, we can better rank the relevant questions, answers and users for a certain query. By studying the tightly connected relationships among different types of objects, we can gain useful insights toward solving the co-ranking problem. However, this is a challenging task due to the following reasons:

- Q&A sites are getting larger with millions and billions of objects. For example, Yahoo! Answers had already hit 1 billion answers in May 2010 [17] and it had received 300 million questions in July, 2012 [18]. So it is important to design a model that not only can rank each type of object well, but also can accurately scale to large websites.

- Another challenge lies in the fact that little prior knowledge about Q&A sites exists. Previous approaches [6, 24] focus on learning to rank different objects in Q&A sites under supervised or semi-supervised settings, which explicitly or implicitly assume the availability of some prior knowledge. However, with large volumes of new questions, answers and users, extracting supervised information from Q&A sites can be very expensive and time consuming.

In order to address these issues, we model a Q&A site as a heterogeneous network, and capture the interdependent relationships to infer the *popularity of questions*, *interestingness of answers* and *contributions of users* simultaneously. Take the ranking of questions for example, Yahoo! Answers simply ranks the questions according to the number of answers they have. However, we observe in Section 3 that the popularity of a question does not depend on how many answers it attracts but on the interestingness of those answers, which is a common heuristic similar to the PageRank [7] principle. For instance, a question ("**Is 1+1=2?**") attracting 100 boring answers ("**Yes**" or "**No**") is much less popular than a question ("**Why do cats act like monkeys?**") attracting 10 interesting answers. We consider such interdependencies in our co-ranking model.

In summary, our contributions can be summarized as follows:

- We explore the interdependent relationships among questions, answers and users, and verify the existence of such relationships in Q&A sites. Based on these interrelationships, we rank each object under an unsupervised setting. To the best of our knowledge, this is the first work that investigates the usefulness of such relationships in Q&A sites.

- We propose an unsupervised **N**etwork-based **C**o-**R**anking model (NCR) to simultaneously recognize high quality objects in Q&A sites. We design the NCR framework in a divide-and-conquer way to decompose the co-ranking problem into three separate types of submodules for questions, answers and users. The interdependencies are captured through iterative computations. To the best of our knowledge, this is the first unsupervised method that solves this problem in Q&A sites.

- We show how the proposed NCR model can be executed in a parallelized environment to scale up to very large Q&A sites. The divide-and-conquer framework makes the co-ranking problem parallelizable. At each iteration, ranking results for users, questions and answers are computed in parallel. Then these results are used in the next iteration to capture interdependencies among objects. The distributed algorithm helps Q&A sites rapidly identify high quality objects even when the data arrives in large volumes.

- We conduct extensive empirical studies on real Yahoo! Answers datasets to demonstrate the effectiveness of the proposed NCR method and the efficiency of the distributed version.

2. PROBLEM DEFINITION

In this section, we first introduce several related concepts and notations. Then, we will formally define the problem of co-ranking multiple types of objects in Q&A networks.

Definition 1. **Heterogeneous Q&A Network**: A heterogeneous Q&A network is a special kind of information network, which is represented as a graph $G = (V, E)$. V is the set of nodes (objects), including t types of objects $T_1 = \{v_{11}, ..., v_{1n_1}\}, ..., T_t = \{v_{t1}, ..., v_{tn_t}\}$. $E \subseteq V \times V$ is the set of links (relations) between the nodes in V, which involves multiple types of links. The network of Yahoo! Answers is shown in Figure 1. It involves three types of objects, i.e., users (U), questions (Q) and answers (A), and three types of links, i.e., *askedBy*, *answeredBy* and *givenBy*.

In heterogeneous Q&A networks, each type of link represents an unique binary relation R from node type i to node type j, where $R(v_{ip}, v_{jq})$ holds iff object v_{ip} and v_{jq} are related by relation R. R^{-1} denotes the inverted relation of R, which holds naturally for $R^{-1}(v_{jq}, v_{ip})$. Let $dom(R) = \mathcal{T}_i$ denote the domain of relation R, $range(R) = \mathcal{T}_j$ denotes its range. For example, in Figure 1, the link type "*askedBy*" can be written as a relation R between question nodes and user nodes. $R(v_{ip}, v_{jq})$ holds iff question v_{ip} is asked by user v_{jq}. Given a Q&A network, we can define different adjacent matrices according to the different types of links as follows:

Definition 2. $\mathbf{L_{QA}}$ **matrix**: $E_{QA} \subseteq Q \times A \subset E$ is the set of *answeredBy* links between questions and answers. The corresponding adjacent matrix can be denoted as L_{QA}, where $L_{QA}(q, a) = 1$ if a question $q \in Q$ has an answer $a \in A$, and $L_{QA}(q, a) = 0$ otherwise. Q and A have the relationship of $1 : n$, i.e., a question may have n answers ($n \geq 1$) but an answer is only for a certain question.

Definition 3. $\mathbf{L_{AU}}$ **matrix**: $E_{AU} \subseteq A \times U \subset E$ is the set of *givenBy* links between answers and users. The corresponding adjacent matrix can be represented as L_{AU}, where $L_{AU}(a, u) = 1$ if an answer $a \in A$ is posted by a user $u \in U$, and $L_{AU}(a, u) = 0$ otherwise. A and U have the relationship of $n : 1$, i.e., an answer is only from a certain user but a user may provide n answers ($n \geq 1$) to different questions.

Definition 4. $\mathbf{L_{QU}}$ **matrix**: $E_{QU} \subseteq Q \times U \subset E$ is the set of *askedBy* links between questions and users. The corresponding adjacent matrix is L_{QU}, where $L_{QU}(q, u) = 1$ if a question $q \in Q$ is asked by a user $u \in U$, and $L_{QU}(q, u) = 0$

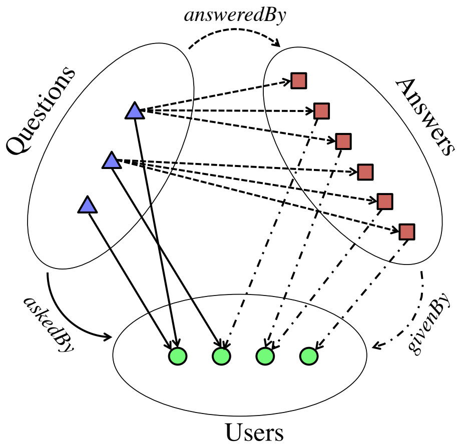


Figure 1: The heterogeneous network in Yahoo! Answers. Each shape represents one type of object in the network, and each arrowed line represents one type of link.

otherwise. Q and U have the relationship of $n:1$, i.e., a question is only asked by a certain user but a user may post n questions ($n \geq 1$) on Q&A websites.

Given a Q&A network, our goal is to rank different types of nodes simultaneously. So we define variables that quantify the qualities of questions, answers and users more precisely:

Definition 5. **Popularity of questions**: The popularity of a question $q \in Q \subset V$ (denoted by $\mathcal{P}(q)$) is a score of how popular the question q is. It indicates the question q's ability to attract hot debates or discussions. For ease of understanding and computations, we limit the range of $\mathcal{P}(q)$ to (0, 1).

For example, "Why do cats act like monkeys?" is a popular question because everyone can be involved to show diverse answers from different viewpoints.

Definition 6. **Interestingness of answers**: The interestingness of an answer $a \in A \subset V$ (denoted by $\mathcal{I}(a)$) is a score of how interesting the answer a is. The interestingness represents the answer a's strength to impress users. $\mathcal{I}(a)$ is within the range (0, 1).

We still take the question "Why do cats act like monkeys?" as an example, an answer "They are scared little cats and hiding from the big dogs." is more interesting than another answer "Wait.....what?".

Definition 7. **Contribution of users**: The contribution of a user $u \in U \subset V$ (denoted by $\mathcal{C}(u)$) is a score of how the user u contributes to the Q&A community. It indicates the user u's ability to ask popular questions and give interesting answers. $\mathcal{C}(u)$ is also within the range (0, 1).

For instance, a user asking 10 popular questions contributes more than one who asks 100 questions with nobody answering them. Similarly, a user providing many interesting answers also contributes a lot to the Q&A system.

Table 1: Statistics of Yahoo! Answers datasets

Subcategory	#questions	#answers	#users	#stars	#votes
All	169,103	1,380,082	263,512	53,035	577,710
Religion & Spirituality	15,926	187,627	22,883	7,318	72,360
Politics & Government	8,823	94,374	11,722	3,269	44,261
Baby Names	3,913	44,687	12,380	1,089	13,831
Dogs	3,291	25,076	9,703	218	11,223

Based on the node and relation types in Q&A networks, the input of the co-ranking task consists of a heterogeneous network $G = (V, E)$. V includes three types of nodes Q, A and U. E involves three types of links E_{QA}, E_{AU} and E_{QU}. Our goal is to build a general framework to simultaneously recognize high quality questions, answers and users under an unsupervised setting. In order to solve this problem, we need to address the following challenges:

1. How can one capture the interdependent relationships among a question's popularity, an answer's interestingness and a user's contribution?
2. Based upon the interdependent relationships, how can one simultaneously rank the questions, answers and users in an unsupervised way?
3. As the network is getting larger, how can one scale up the ranking algorithm to the growth of Q&A sites?

For the first challenge, we explore some observations on the real Yahoo! Answers data. Based on them, we present the interdependent relationships among the three different types of nodes in Section 3. For the second challenge, we introduce how to iteratively rank the three types of nodes based on the interrelationships in Section 4. For the third challenge, we implement the framework in a parallelized environment and show the time efficiency in Section 5.

3. DATA ANALYSIS

A motivation for this work is that a question's popularity, an answer's interestingness, and a user's contribution could be strongly correlated with each other in Q&A websites. Before proceeding, we first introduce real-world data used in this work and investigate whether the observations support the assumption of interdependent relationships among questions, answers and users.

3.1 Data Description

Yahoo! Answers is a popular Q&A website where people ask and answer questions on any topic. Each question has a lifecycle. After it is posted by an asker, it stays in an "open" state where it receives answers. Then at some point (decided by the asker, or by an automatic timeout on the website), the question is considered "closed" and can receive no further answers. At this stage, a "best answer" is selected either by the asker or through a voting procedure from other users; once a best answer is chosen, the question is "resolved". A question can also be awarded a "star" by any user at any time, marking it as an interesting question.

We collect 169,103 resolved questions posted in April, 2013. Each of these questions has at least 5 answers and the "best answer" is selected according to the votes of other users. We also randomly select 4 subcategories and take the questions, answers and users under each subcategory as a small dataset. The data is publicly available on Yahoo! Answers and the basis statistics are shown in Table 1.

3.2 Verifying Interdependent Relationships

In this section, we conduct a data analysis on questions, answers and users under the subcategory of Religion & Spirituality. We investigate some of the basic principles that

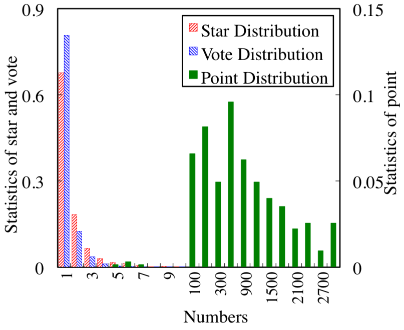


Figure 2: Distributions of question stars, answer votes and user points on Yahoo! Answers. Question stars reflect the popularity of questions, answer votes show the interestingness of answers, and user points indicate the contribution of users to some extent.

reveal the potential interdependent relationships among the three types of nodes in the heterogeneous Q&A network.

On Yahoo! Answers, a user can award a question a "star", marking it as an interesting question. The number of stars can reflect the popularity of a question. The red bars in Figure 2 indicate that about 97% of the questions have at most 5 stars. A user can also vote an answer as a "best answer". The number of votes can reveal the interestingness of an answer. The blue bars in Figure 2 show that around 98% of the answers have at most 3 votes. A user can be awarded more points if s/he provides more answers. The number of points can reflect the contribution of a user to some extent. We can observe that around 60% of the users get points less than 3000 from the green bars in Figure 2.

Based on these statistics, we first conduct a data analysis on questions. Figure 3 (a) reveals the influences of answers and users on questions. It can be observed that if a question has more stars, its answers will have more votes and the asker will also have more points. This phenomenon illustrates that

(1) Popular questions often attract more interesting answers than those questions with little popularity. (2) Popular questions are likely to be asked by those high contribution users.

Then we analyze the influences of questions and users on answers. Figure 3 (b) shows the analysis. If an answer has more votes, the corresponding question usually has more stars and the user who provides the answer often has more points. This observation shows that interesting answers are usually given to those popular questions and they are often provided by those high contribution users.

We also do some analysis on the users with no more than 3000 points. Figure 3 (c) reveals the influences of questions and answers on users. If a user has more points, his/her questions often have more stars and his/her answers often have more votes. It illustrates that high contribution users would like to ask popular questions and post interesting answers.

The relationship between answers and users we observed is consistent with that in [3]. Furthermore, we additionally capture the relationships between a question and its asker. Based on these observations, we propose the following principles to solve the co-ranking problem in Q&A networks:

1. We can identify the interestingness of an answer given the popularity of the corresponding question, plus the contribution of the user who posted the answer.
2. For a question, if we have the interestingness scores of its answers and the contribution score of its asker, we can infer its popularity, because a question is more popular if it attracts more interesting answers and the asker has more contribution to the system.
3. Now we go back to indicate a user's contribution. Intuitively, a user contributes more to the community if s/he asks more popular questions and gives more interesting answers. In contrast, one contributes less if one has few popular questions and interesting answers.

With these observations and principles, we next introduce how to model the qualities of different types of nodes in the network-based co-ranking framework.

4. PROPOSED METHOD

In this section, we propose an unsupervised **N**etwork-based **C**o-**R**anking model (**NCR**) and introduce the iterative computation algorithm for NCR. The above observations and principles serve as the base of the proposed model. NCR is designed in a divide-and-conquer way to decompose the co-ranking problem into three separate types of submodules (as shown in Figure 4) for questions, answers and users. The interdependencies are captured through iterative computations to help identify high quality objects in heterogeneous Q&A networks.

4.1 Question Popularity

Given a question q, we denote the set of answers for q as $\mathcal{S}_a(q) = \{a_i \mid \forall a_i, L_{QA}(q, a_i) = 1\}$. The influence of $\mathcal{S}_a(q)$ on the popularity of q is defined as the summation of the interestingnesses of all the answers in $\mathcal{S}_a(q)$,

$$\Im_a(q) = \frac{\sum_{k=1}^{|\mathcal{S}_a(q)|} \mathcal{I}(a_k)}{\mathcal{N}_{aq}} \tag{1}$$

where $\mid \mathcal{S}_a(q) \mid$ is the number of answers q has. We use $\mathcal{N}_{aq}$ to normalize $\Im_a(q)$ on the entire question set Q so their squares sum to 1: $\sum_{q \in Q} (\Im_a(q))^2 = 1$.

Similarly, the set of users for q can be denoted as $\mathcal{S}_u(q) = \{u_i \mid \forall u_i, L_{QU}(q, u_i) = 1\}$. Since each question can only be asked by a certain user, we can denote the only element as u_q. The influence of u_q on the popularity of q can be defined in a similar way as follows:

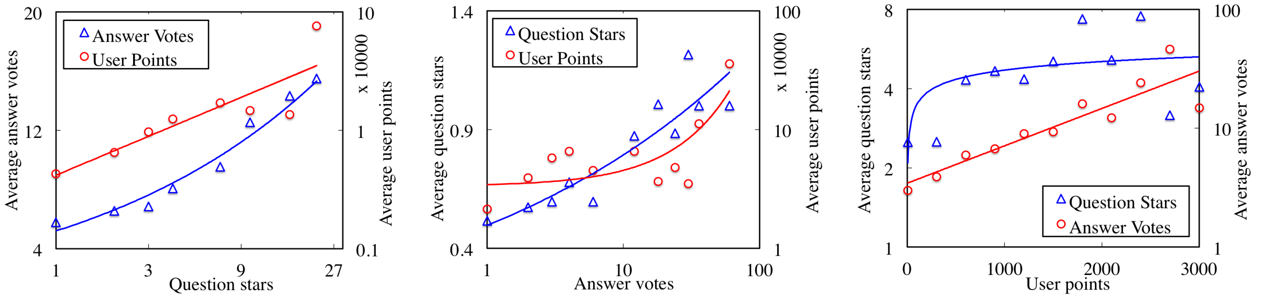
$$\Im_u(q) = \frac{\mathcal{C}(u_q)}{\mathcal{N}_{uq}} \tag{2}$$

where $\mathcal{N}_{uq} = \sum_{q \in Q} (\Im_u(q))^2$ is a normalization factor on the entire question set Q so their squares sum to 1.

Depending on the influences of answers and users, we can compute the popularity of q as follows:

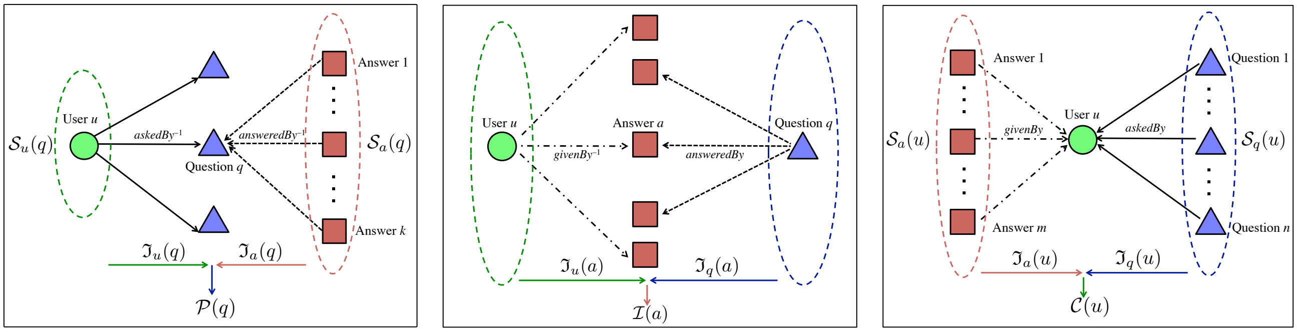
$$\mathcal{P}(q) = \frac{\Im_a(q) + \Im_u(q)}{\mathcal{N}_q} \tag{3}$$

where $\mathcal{N}_q = \sum_{q \in Q} (\mathcal{P}(q))^2$ is a normalization factor on the entire question set Q so their squares sum to 1. In Equation 3, we add $\Im_a(q)$ and $\Im_u(q)$ together instead of multiplying them. The reason is that if a question is newly posted without any answers, its popularity can still be predicted by its asker's contribution. However, if we multiply $\Im_a(q)$ and $\Im_u(q)$ together, the popularity becomes 0, which will cause



(a) Influence on questions (b) Influence on answers (c) Influence on users

Figure 3: Interdependent relationships among questions, answers and users on Yahoo! Answers. Each line shows the trend of influence one type of object on another type.



(a) Calculation of $\mathcal{P}(q)$ (b) Calculation of $\mathcal{I}(a)$ (c) Calculation of $\mathcal{C}(u)$

Figure 4: Calculations of question popularity, answer interestingness and user contribution in NCR model. The colored ellipse represents the set of nodes that are interconnected with the node we are focusing on.

inaccurate rankings for new questions on Q&A sites. Figure 4 (a) summaries the entire process of computing the popularity score of a question q according to the influences of its answers and user.

In order to calculate $\mathcal{P}(q)$, we need the interestingness values of q's answers, which we define next.

4.2 Answer Interestingness

To decide the interestingness of an answer, we have similar intuitions to the calculation of question popularity. Given an answer a, the answerer u belongs to U ($u \in U$) and $L_{AU}(a,u) = 1$. Similarly, the corresponding question q belongs to Q ($q \in Q$) and $L_{QA}(q,a) = 1$. We denote the influences of u and q on the interestingness of a as $\mathfrak{I}_u(a)$ and $\mathfrak{I}_q(a)$, respectively. According to Figure 3 (b), the interestingness of an answer is highly influenced by its question's popularity and its user's contribution. Figure 4 (b) represents how to obtain the interestingness of a via the influences of q and u.

$\mathfrak{I}_u(a)$, $\mathfrak{I}_q(a)$ and $\mathcal{I}(a)$ can be formulated in a similar way.

$$\mathfrak{I}_u(a) = \frac{\mathcal{C}(u_a)}{\mathcal{N}_{ua}} \tag{4}$$

$$\mathfrak{I}_q(a) = \frac{\mathcal{P}(q_a)}{\mathcal{N}_{qa}} \tag{5}$$

$$\mathcal{I}(a) = \frac{\mathfrak{I}_u(a) + \mathfrak{I}_q(a)}{\mathcal{N}_a} \tag{6}$$

where $\mathcal{N}_{ua}$, $\mathcal{N}_{qa}$ and $\mathcal{N}_a$ are three normalization factors to ensure the scale to be within the range (0, 1).

4.3 User Contribution

How do we judge a user? We usually consider two things when we look at a user. The first one is the answers the user provides. If the answers are very interesting and most of them are selected as "best answers", the user tend to contribute much to the Q&A community. In contrast, if the answers are always boring (like "Yes", "No" or "What?"), we may doubt the contribution of the user. The second factor is the questions the user asks. Those users are likely to contribute more if their questions are popular with lots of interesting answers. Based on these observations, we model a user's contribution by considering both his/her answers and questions. Figure 4 (c) shows how to calculate the contribution score of a user via the influence of his/her answers and questions.

Given a user u, we denote the set of answers u gives as $\mathcal{S}_a(u) = \{a_i \mid \forall a_i, L_{AU}(a_i,u) = 1\}$. We formulate the influence of $\mathcal{S}_a(u)$ on the contribution of u as the summation of the interestingness scores of all answers in $\mathcal{S}_a(u)$,

$$\mathfrak{I}_a(u) = \frac{\sum_{m=1}^{|\mathcal{S}_a(u)|} \mathcal{I}(a_m)}{\mathcal{N}_{au}} \tag{7}$$

where $\mid \mathcal{S}_a(u) \mid$ is the number of answers u provides. We use $\mathcal{N}_{au}$ to normalize $\mathfrak{I}_a(u)$ on the entire user set U so their squares sum to 1: $\sum_{u \in U}(\mathfrak{I}_a(u))^2 = 1$.

Similarly, the set of questions u asks can be denoted as $\mathcal{S}_q(u) = \{q_i \mid \forall q_i, L_{QU}(q_i,u) = 1\}$, and the influence of $\mathcal{S}_q(u)$ on the contribution of u is the summation of the pop-

Algorithm 1 The NCR algorithm

```
Input: A heterogeneous Q&A network $G = (V, E)$, maximum # of iteration $Max\_It$
Output: The set of popularity $\mathcal{P}$, interestingness $\mathcal{I}$ and contribution $\mathcal{C}$
//Initialization step
Initialize question's popularity, answer's interestingness and user's contribution to 1
while NOT converged or #iteration <= $Max\_It$ do
    Update $\mathcal{P}$ using (1), (2) and (3)
    Update $\mathcal{I}$ using (4), (5) and (6)
    Update $\mathcal{C}$ using (7), (8) and (9)
end while
```

ularity scores of all questions in $\mathcal{S}_q(u)$,

$$\mathfrak{I}_q(u) = \frac{\sum_{n=1}^{|\mathcal{S}_q(u)|} \mathcal{P}(q_n)}{\mathcal{N}_{qu}} \tag{8}$$

where $\mid \mathcal{S}_q(u) \mid$ is the number of questions u asks. $\mathcal{N}_{qu} = \sum_{u \in U} (\mathfrak{I}_q(u))^2$ is used to normalize $\mathfrak{I}_a(u)$ on the entire user set U so their squares sum to 1.

Depending on the influences of answers and questions, we can compute the contribution score of u as follows:

$$\mathcal{C}(u) = \frac{\mathfrak{I}_a(u) + \mathfrak{I}_q(u)}{\mathcal{N}_u} \tag{9}$$

where $\mathcal{N}_u = \sum_{u \in U} (\mathcal{C}(u))^2$ is a normalization factor on the entire user set U.

4.4 Iterative Computation Framework

Integrating all the information of the heterogeneous Q&A network together, NCR adopts an iterative method to compute question popularity, answer interestingness and user contribution, by exploring the interdependent relationships among them. The iterative computation framework is summarized in Algorithm 1. The time complexity of the NCR algorithm is $O(t|E|)$, where t is the iteration number and $|E|$ is the number of links in the Q&A network. Through our experiments, the algorithm converges after 3 rounds in most cases.

4.5 Parallel Computing of NCR

The proposed NCR model computes the quality of each object in Q&A sites according to the quality of its linked objects. The very design of NCR makes the entire network naturally splittable when NCR calculates the qualify of objects. As shown in Figure 4, the ranking results for questions, answers and users can be computed in parallel at each iteration. Therefore, by extending NCR to a distributed version, we can reduce the runtime significantly. In the experiments, we choose to implement the distributed version of NCR on Apache Hadoop Platform using Pig Latin[1]. Pig Latin is a high-level programming language. It allows the developer to specify how the algorithm is performed, while the Pig complier transforms the specifications into Map-Reduce programs. The runtime of the Map-Reduce jobs depends on how the data is split and stored on the cluster [20]. In this paper, we test the scalability of the distributed NCR algorithm by generating various numbers of data splits.

5. EXPERIMENTS

In this section, we conduct extensive experiments to evaluate the proposed NCR framework. After introducing the experiment settings and the evaluation metric, we compare different ranking methods. Furthermore, we also study the efficiency of NCR on the large-scale network.

5.1 Experimental Setup

We test the effectiveness of the proposed NCR model on datasets showed in Table 1. In order to evaluate the performance, we have to generate ranking lists for questions, answers and users respectively from real Yahoo! Answers datasets. Since [13] considers stars as one metric for question quality, we use stars as the ground truth for question popularities. Wang et al. [29] use "best answer" labels to evaluate the quality of answers. However, only one "best answer" is selected for a question and all the other answers have the same labels. It will be biased to evaluate our ranking results using such labels. Since a "best answer" is selected through a voting procedure from other users, we use vote numbers as the ground truth for answer interestingness in our experiments. For users, we generate the ground truth of a user by considering both the stars of questions s/he asked and the votes of answers s/he provided because they reflect the contribution of users in [11]. With such rich "human labelings" on Yahoo! Answers, we average these values and then convert them into integers in a scale from level 1 (the lowest quality) to level 4 (the highest quality) as in [13].

We set up the evaluation criterion using normalized discounted cumulative gain (nDCG) [9]. nDCG is a popular measure in information retrieval tasks, and it focuses on correct rankings of high quality nodes. We compare the result of NCR with the ground truth for each type of node. For questions and users, we can obtain a single ranking list with the entire participating nodes and then calculate the nDCG values. However, it is different for the answers, since an answer is only useful to its corresponding question. If we rank all the answers together, it would be meaningless. So for each question, we rank its answers and calculate the nDCG value. Then we use the average nDCG value as the final evaluation of answer quality.

5.2 Compared Methods

In order to demonstrate the effectiveness of our NCR approach, we compare the following methods:

- PR: This method is proposed in [32] which uses PageRank algorithm [7] to rank the contribution of users. A directed post-reply network is constructed by viewing each user as a node and linking the asker to everyone who replied to his/her questions. Note a user may answer more than one question posted by another user, the frequencies one replies another are considered as weights to edges as in [32]. Since it is difficult to create a meaningful homogeneous network of questions (answers), we only run this method on the ranking task of user contribution.

- HITS (A): This baseline method is also used in [32]. It ranks the contribution of users according to the authority value of HITS algorithm [12]. In the post-reply network of users, a good authority is a user who helps many good askers (hubs) and s/he may have high con-

[1] http://pig.apache.org/

tribution by providing useful answers according to Definition 7.

- HITS (H): We compare with another baseline using HITS algorithm. We use hub values of HITS to correspond to contribution ranks of users. A good hub in the post-reply network is a user who is helped by many good answerers (authorities). Such user may also have high contribution by asking popular questions according to Definition 7.

- HITS (UQ): This method is extended from HITS to rank both the users and questions in a heterogeneous network containing users and the questions they answered. If a user u_i answered a question q_j, we add a link from u_i to q_j. A good authority is a popular question that attracts many good users (hubs) to answer, and a good hub is a user who provides answers to many good questions (authorities). We can only generate such a heterogeneous network with two different types of nodes since users and the questions they answered are n to n relationships while any other two types of nodes are not.

- RAT: This baseline method is the **R**anking of **A**nswer **T**ime (RAT) for answers' interestingness. It is derived from [3], which shows that high quality answers often arrive earlier. RAT algorithm ranks the interestingness of answers for a question according to their posting time.

- RCS: This method is the **R**anking of **C**osine **S**imilarity (RCS) for answers' interestingness. It is based on the assumption that a high quality answer and the corresponding question often focus on similar content. So we rank each answer according to its cosine similarity with its question.

- RAD: This method is the **R**anking of **A**verage **D**iversity (RAD) for questions' quality. We assume that a popular question usually attracts many diverse answers. We define the diversity between two answers as $div(a_i, a_j) = 1 - cosine(a_i, a_j)$. A larger value means greater diversity between two answers of a question. RAD algorithm ranks the popularity of each question according to the average diversity on all of its answer pairs.

All these baseline methods ignore the interdependent relationships in the heterogeneous Q&A network (e.g., PR, CSR, etc.) or only consider incomplete relationships (e.g., HITS(UQ)).

5.3 Performance Evaluation

In this subsection, we study the effectiveness of the proposed NCR method. Table 2 presents the comparison results for question popularity, answer interestingness and user contribution under the subcategory of Religion & Spirituality. Due to space limit, we only show the performances of question popularity and user contribution from top 10% to 50% in Table 2 (a) and (c). Since each question has at least 5 answers, we report the average nDCG value from top 1 to top 5 for interestingness of answers in Table 2 (b). It can be observed that NCR consistently outperforms other baseline methods on the rankings of all the three types of nodes.

In particular, compared with the three baseline methods PR, HITS (A) and HITS (H), all of which are focusing on the homogeneous network of users, NCR can achieve the best performance as shown in Table 2 (c). It illustrates that the interdependent relationships among questions, answers and users can help improve the ranking result of users. Moreover, although PR and HITS (A) can identify high contribution users with good results, NCR can still perform better than these two baseline methods with an improvement of at least 3.5%. Furthermore, the performance of NCR is significantly better than that of HITS (H) with an improvement of at most 24%.

Compared with the baseline method HITS (UQ) considering the heterogeneous network with two types of nodes (questions and users), NCR can still have better performances on both questions and users as shown in Table 2 (a) and (c). It reveals that questions, answers and users are tightly interconnected with each other and ignoring either one type of object would weaken the co-ranking performance.

Compared with the other baseline methods RAT, RCS and RAD, which do not consider the network structure in the ranking task, NCR still achieves the best results. It shows that the interdependent relationships in the heterogeneous network are more powerful and helpful than the time information (used in RAT) and the text information (used in RCS and RAD) in detecting high quality nodes. Moreover, extracting meaningful features from text information is challenging and time-consuming. With large volumes of new questions and answers, making use of text information efficiently becomes more and more difficult.

We further show the performances of the proposed NCR model under another three subcategories in Tables 3, 4 and 5. We can observe that the performances of NCR are the best under all these subcategories except that of user contribution under the subcategory of Dogs. Though NCR does not perform well for user contribution, it still achieves the best results for question popularity and answer interestingness. Moreover, Table 4 (a) shows that NCR can achieve an improvement of at most 140% on question quality under the subcategory of Baby Names. We also present the result on the entire Yahoo! Answers data in Table 6. It reveals that NCR is robust and stable regardless of any category information.

In summary, with the help of interdependent relationships among questions, answers and users, NCR always outperforms the baseline methods. In the next subsection, we investigate more details about the efficiency of the proposed NCR framework.

5.4 Scalability Performance

In this subsection, we evaluate the runtime efficiency of two versions of NCR: single-NCR runs in Python on a single node Server (Intel Xeon^{TM} Quad-Core CPUs of 2.26GHz and 36GB RAM) and parallel-NCR runs on a multi-node Hadoop cluster (Intel Xeon^{TM} 2×Quad-Core CPUs of 2.50GHz and 16GB RAM) in Pig Latin. Parallel-NCR algorithm is complied into Map-Reduce jobs and executed over Hadoop in a distributed fashion. We use the entire dataset with questions from all categories and report the runtime of these two versions in Table 7. Since the ranking results are the same no matter we run NCR in local or in parallel, we only present the runtime (in seconds) in Table 7. It can be observed that with 8-node Hadoop cluster, parallel-NCR is 6.5 times faster than single-NCR.

Table 2: Ranking performances on Yahoo! Answers data under subcategory of Religion & Spirituality. “↑” indicates that the larger the value the better the performance. The number in bold indicates the best performance.

(a) Question popularity

Criterion	Top k%	Methods		
		HITS (UQ)	RAD	NCR
nDCG ↑	10	0.641	0.640	**0.715**
	20	0.691	0.705	**0.757**
	30	0.718	0.729	**0.766**
	40	0.774	0.780	**0.811**
	50	0.819	0.830	**0.855**

(b) Answer interestingness

Criterion	Top k	Methods		
		RAT	RCS	NCR
Average nDCG ↑	1	0.563	0.581	**0.613**
	2	0.695	0.716	**0.732**
	3	0.748	0.770	**0.781**
	4	0.785	0.806	**0.814**
	5	0.814	0.833	**0.839**

(c) User contribution

Criterion	Top k%	Methods				
		PR	HITS (A)	HITS (H)	HITS (UQ)	NCR
nDCG ↑	10	0.866	0.863	0.721	0.764	**0.897**
	20	0.786	0.776	0.663	0.686	**0.824**
	30	0.782	0.770	0.677	0.695	**0.828**
	40	0.801	0.794	0.722	0.733	**0.850**
	50	0.840	0.834	0.780	0.789	**0.890**

Table 3: Ranking performances on Yahoo! Answers data under subcategory of Politics & Government. “↑” indicates that the larger the value the better the performance. The number in bold indicates the best performance.

(a) Question popularity

Criterion	Top k%	Methods		
		HITS (UQ)	RAD	NCR
nDCG ↑	10	0.634	0.605	**0.688**
	20	0.687	0.672	**0.731**
	30	0.732	0.720	**0.761**
	40	0.788	0.783	**0.818**
	50	0.830	0.830	**0.862**

(b) Answer interestingness

Criterion	Top k	Methods		
		RAT	RCS	NCR
Average nDCG ↑	1	0.569	0.577	**0.599**
	2	0.704	0.717	**0.730**
	3	0.758	0.774	**0.781**
	4	0.795	0.811	**0.816**
	5	0.824	0.838	**0.843**

(c) User contribution

Criterion	Top k%	Methods				
		PR	HITS (A)	HITS (H)	HITS (UQ)	NCR
nDCG ↑	10	0.892	0.900	0.738	0.800	**0.927**
	20	0.859	0.867	0.685	0.756	**0.891**
	30	0.842	0.848	0.699	0.750	**0.873**
	40	0.843	0.847	0.728	0.764	**0.870**
	50	0.880	0.881	0.783	0.809	**0.906**

Table 4: Ranking performances on Yahoo! Answers data under subcategory of Baby Names. “↑” indicates that the larger the value the better the performance. The number in bold indicates the best performance.

(a) Question popularity

Criterion	Top k%	Methods		
		HITS (UQ)	RAD	NCR
nDCG ↑	10	0.580	0.355	**0.857**
	20	0.519	0.396	**0.828**
	30	0.520	0.476	**0.844**
	40	0.594	0.557	**0.884**
	50	0.673	0.608	**0.907**

(b) Answer interestingness

Criterion	Top k	Methods		
		RAT	RCS	NCR
Average nDCG ↑	1	0.548	0.581	**0.637**
	2	0.700	0.733	**0.767**
	3	0.758	0.792	**0.819**
	4	0.797	0.829	**0.851**
	5	0.827	0.855	**0.874**

(c) User contribution

Criterion	Top k%	Methods				
		PR	HITS (A)	HITS (H)	HITS (UQ)	NCR
nDCG ↑	10	0.766	0.762	0.675	0.676	**0.820**
	20	0.740	0.730	0.662	0.655	**0.776**
	30	0.804	0.794	0.732	0.723	**0.829**
	40	0.847	0.841	0.792	0.780	**0.872**
	50	0.877	0.873	0.837	0.827	**0.903**

Table 5: Ranking performances on Yahoo! Answers data under subcategory of Dogs. “↑” indicates that the larger the value the better the performance. The number in bold indicates the best performance.

(a) Question popularity

Criterion	Top k%	Methods		
		HITS (UQ)	RAD	NCR
nDCG ↑	10	0.586	0.630	**0.688**
	20	0.637	0.705	**0.728**
	30	0.709	0.767	**0.804**
	40	0.755	0.809	**0.831**
	50	0.801	0.828	**0.857**

(b) Answer interestingness

Criterion	Top k	Methods		
		RAT	RCS	NCR
Average nDCG ↑	1	0.573	0.607	**0.620**
	2	0.714	0.749	**0.776**
	3	0.773	0.810	**0.835**
	4	0.816	0.849	**0.873**
	5	0.852	0.879	**0.897**

(c) User contribution

Criterion	Top k%	Methods				
		PR	HITS (A)	HITS (H)	HITS (UQ)	NCR
nDCG ↑	10	**0.761**	0.732	0.449	0.664	0.678
	20	**0.743**	0.719	0.498	0.663	0.680
	30	**0.795**	0.777	0.581	0.727	0.755
	40	**0.842**	0.827	0.665	0.784	0.816
	50	**0.876**	0.865	0.737	0.829	0.857

Table 6: Ranking performances on Yahoo! Answers data for all topic categories. “↑” indicates that the larger the value the better the performance. The number in bold indicates the best performance.

(a) Question popularity

Criterion	Top k%	Methods		
		HITS (UQ)	RAD	NCR
nDCG ↑	10	0.661	0.657	**0.710**
	20	0.739	0.719	**0.766**
	30	0.771	0.742	**0.788**
	40	0.835	0.805	**0.850**
	50	0.880	0.854	**0.894**

(b) Answer interestingness

Criterion	Top k	Methods		
		RAT	RCS	NCR
Average nDCG ↑	1	0.574	0.597	**0.618**
	2	0.723	0.750	**0.764**
	3	0.785	0.811	**0.821**
	4	0.827	0.850	**0.858**
	5	0.860	0.878	**0.885**

(c) User contribution

Criterion	Top k%	Methods				
		PR	HITS (A)	HITS (H)	HITS (UQ)	NCR
nDCG ↑	10	0.844	0.766	0.632	0.793	**0.856**
	20	0.807	0.758	0.616	0.774	**0.822**
	30	0.801	0.769	0.626	0.779	**0.817**
	40	0.833	0.812	0.672	0.818	**0.850**
	50	0.872	0.860	0.747	0.864	**0.892**

We also test the scalability of parallel-NCR with different mappers. We set the data split size (mega byte) to be 6 different numbers from 5 to 30, with an interval of 5. The smaller the data split size is, the larger the number of mappers is. Figure 5 summarizes the performance of mapping time with different numbers of mappers. It can be seen that with more mappers, the mapping process becomes much more efficient. For example, if 70 mappers are used to run parallel-NCR, the mapping procedure costs 324 seconds. If we use more mappers such as 129 mappers, the mapping

Table 7: Scalability performance of NCR.

Methods	Implementation	# of nodes	Runtime (seconds)
Single-NCR	Python	1	11,969
Parallel-NCR	Pig Latin	8	1,593

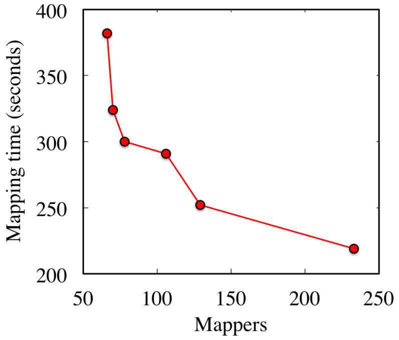


Figure 5: The mapping time with different number of mappers.

time is reduced to 252 seconds. However, the communication time will increase if we use more mappers. So in our experiment, we set the data split size as 10Mb. The total number of mappers are 129.

In summary, as the network gets larger, parallel-NCR can efficiently scale up the proposed framework to the growth of Q&A sites.

6. RELATED WORK

User communities in online Q&A websites have already been investigated from several perspectives. The first is the study of user's interests and motivations for contribution [1, 21, 4, 27] in the community. These studies model the authority, reputation and expertise of users in social networks and communities [8, 16, 15]. The second is the study of user quality in Q&A sites by developing several link-based ranking algorithms [11, 19, 30, 32]. For example, Zhang et al. [32] focus on the data from Java forum and construct a post-reply network, in which the nodes correspond to users and the links represent interactions between askers and answerers. Both ExpertiseRank (a PageRank-like algorithm) and HITS are applied in [32] to identify users with high expertise. Jurczyk and Agichtein [11] also apply the HITS algorithm to user communities and they aim to discover authoritative users in topical categories. However, all these studies focus on ranking only users by extracting a homogeneous user network from Q&A sites. Our study is different from them since we model the Q&A site as a heterogeneous network and co-rank questions, answers and users in this network.

The study of content quality in Q&A websites is also related to our work. It can be categorized into two branches. The first branch investigates how to evaluate or predict the question quality [2, 13, 14]. Agichtein et al. [2] analyze the essential features related to questions and propose a supervised method to identify high quality content in Q&A sites. [13] evaluates the question quality using a mutual reinforcement-based label propagation algorithm. [14] predicts the subjectivity of questions based on a co-training model. The other branch aims to find high quality answers [10, 5, 29, 26]. Jeon et al. [10] extract non-textual features to predict the quality of answers. Sekai et al. [22] apply the graded-relevance metrics to evaluate the answer quality. Bian et al. [5] introduce a ranking algorithm to retrieve high quality answers according to the user interaction and the answer relevance. Wang et al. [29] model the question-answer relationships via analogical reasoning and develop an answer ranking method. Suryanto et al. [26] aim to find good answers for newly-arrived questions by considering the answerer expertise. However, all these works do not study the quality of users. In addition, some of them [2, 13, 29, 14] require substantial amounts of manual supervision.

There are also a few research studies on the link-based ranking within heterogeneous networks. Studies in [33, 25] focus on the bibliographic data with three different types of objects, authors, publications and conferences. Zhou et al. [33] uses the heterogeneous network of authors and publications to co-rank these two types of objects while [25] ranks authors and conferences by constructing another heterogenous network. Yin et al. [31] create a heterogeneous network of facts and websites to help discover truth in multiple conflicting sources on the Web. Wang et al. [28] detect review spammers based on a heterogeneous online store review network. However, the network representations in these studies are very different from that of the Q&A site, therefore their techniques are not applicable to our work. Bian et al. [6] utilize the relationships among questions, answers and users to estimate the quality of these three types of objects. It is most related to our work. However, [6] uses a semi-supervised method while our work aims to rank the different types of objects in a totally unsupervised way. Another difference is that [6] separates the contribution of users into two groups, the contribution of askers and those of the answerers. But this separation is not realistically reasonable. User contributions should be considered as integrations of their question asking and answering behaviors. In our work, we quantify the contribution of users according to both their asking and answering activities.

7. CONCLUSION

In this paper, we study the problem of estimating the quality of questions, answers and users simultaneously in heterogeneous Q&A networks. First we analyze real-world Yahoo! Answers datasets to demonstrate the interdependent relationships among questions, answers and users. The NCR framework is then introduced to co-rank different types of objects by capturing their interrelationships in an unsupervised way. Extensive experiments are conducted to evaluate the proposed NCR framework on datasets under different topic categories. The effectiveness of the proposed model shows that interdependent relationships play important roles in ranking objects (questions, answers and users) in a heterogeneous Q&A network. Furthermore, we also implement the NCR framework in a distributed version to test its scalability.

There are several interesting directions for future work. In current work, we recognize high quality questions, answers and users based on the interdependent relationships in heterogeneous Q&A networks. Since questions and answers have contents, one direction of our future work is to combine the content information with the interconnections to better rank different types of objects. Some previous studies demonstrate that the community activity plays an important role in Q&A sites. Therefore, another direction of our future work is to add the community information into the proposed framework. Since ranking is an important signal for search, we can also incorporate the co-ranking results into answer vertical search.

Acknowledgement

This work is supported in part by NSF through grants CNS-1115234 and OISE-1129076, US Department of Army through grant W911NF-12-1-0066, and Yahoo! Labs Faculty Research and Engagement Program (FREP).

8. REFERENCES

[1] L. Adamic, J. Zhang, E. Bakshy, and M. Ackerman. Knowledge sharing and yahoo answers: everyone knows something. In *Proc. WWW*, pages 665–674, 2008.

[2] 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.

[3] A. Anderson, D. Huttenlocher, J. Kleinberg, and J. Leskovec. Discovering value from community activity on focused question answering sites: a case study of stack overflow. In *Proc. KDD*, pages 850–858, 2012.

[4] C. Aperjis, B. Huberman, and F. Wu. Human speed-accuracy tradeoffs in search. In *Proc. HICSS*, pages 1–10, 2011.

[5] J. Bian, Y. Liu, E. Agichtein, and H. Zha. Finding the right facts in the crowd: factoid question answering over social media. In *Proc. WWW*, pages 467–476, 2008.

[6] 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.

[7] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. *Computer Networks and ISDN Systems*, 30(1):107–117, 1998.

[8] G. Dror, D. Pelleg, O. Rokhlenko, and I. Szpektor. Churn prediction in new users of yahoo! answers. In *Proc. WWW Companion*, pages 829–834, 2012.

[9] K. Jarvelin and J. Kekalainen. Cumulated gain-based evaluation of ir techniques. *TOIS*, 20(4):422–446, 2002.

[10] 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.

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

[12] J.M. Kleinberg. Authoritative sources in a hyperlinked environment. *JACM*, 46(5):604–632, 1999.

[13] B. Li, T. Jin, M. Lyu, I. King, and B. Mak. Analyzing and predicting question quality in community question answering services. In *Proc. WWW Companion*, pages 775–782, 2012.

[14] B. Li, Y. Liu, and E. Agichtein. Cocqa: co-training over questions and answers with an application to predicting question subjectivity orientation. In *Proc. EMNLP*, pages 937–946, 2008.

[15] Q. Liu, E. Agichtein, G. Dror, E. Gabrilovich, Y. Maarek, D. Pelleg, and I. Szpektor. Predicting web searcher satisfaction with existing community-based answers. In *Proc. SIGIR*, pages 415–424, 2011.

[16] Q. Liu, E. Agichtein, G. Dror, Y. Maarek, and I. Szpektor. When web search fails, searchers become askers: understanding the transition. In *Proc. SIGIR*, pages 801–810, 2012.

[17] M. McGee. Yahoo answers hits one billion answers. *Search Enginel Land*, 2010.

[18] M. McGee. Yahoo answers hits 300 million questions, but q&a activity is declining. *Search Enginel Land*, 2012.

[19] K. Nam, M. Ackerman, and L. Adamic. Questions in, knowledge in?: a study of naver's question answering community. In *Proc. CHI*, pages 779–788, 2009.

[20] C. Olston, B. Reed, U. Srivastava, R. Kumar, and A. Tomkins. Pig latin: a not-so-foreign language for data processing. In *Proc. SIGMOD*, pages 1099–1110, 2008.

[21] J. Preece, B. Nonnecke, and D. Andrews. The top five reasons for lurking: improving community experiences for everyone. *Computers in human behavior*, 20(2):201–223, 2004.

[22] 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*, pages 187–196, 2011.

[23] C. Shah and J. Pomerantz. Evaluating and predicting answer quality in community qa. In *Proc. SIGIR*, pages 411–418, 2010.

[24] X. Si, E. Y. Chang, Z. Gyöngyi, and M. Sun. Confucius and its intelligent disciples: integrating social with search. *PVLDB*, 3(1-2):1505–1516, 2010.

[25] Y. Sun, J. Han, P. Zhao, Z. Yin, H. Cheng, and T. Wu. Rankclus: integrating clustering with ranking for heterogeneous information network analysis. In *Proc. EDBT*, pages 565–576, 2009.

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

[27] G. Wang, K. Gill, M. Mohanlal, H. Zheng, and B. Zhao. Wisdom in the social crowd: an analysis of quora. In *Proc. WWW*, pages 1341–1352, 2013.

[28] G. Wang, S. Xie, B. Liu, and P. Yu. Review graph based online store review spammer detection. In *Proc. ICDM*, pages 1242–1247, 2011.

[29] X. J. Wang, X. Tu, D. Feng, and L. Zhang. Ranking community answers by modeling question-answer relationships via analogical reasoning. In *Proc. SIGIR*, pages 179–186, 2009.

[30] J. Yang, L. Adamic, and M. Ackerman. Crowdsourcing and knowledge sharing: strategic user behavior on taskcn. In *Proc. EC*, pages 246–255, 2008.

[31] X. Yin, J. Han, and P. Yu. Truth discovery with multiple conflicting information providers on the web. *TKDE*, 20(6):796–808, 2008.

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

[33] D. Zhou, S. Orshanskiy, H. Zha, and L. Giles. Co-ranking authors and documents in a heterogeneous network. In *Proc. ICDM*, pages 739–744, 2007.