Ranking Community Answers by Modeling Question-Answer Relationships via Analogical Reasoning

Xin-Jing Wang
Microsoft Research Asia
4F Sigma, 49 Zhichun Road
Beijing, P.R.China
xjwang@microsoft.com

Xudong Tu*, Dan Feng
Huazhong Sci.&Tech. Univ.
1037 Luoyu Road, Wu Han
Hu Bei, P.R.China
{tuxudong,dfeng}@hust.edu.cn

Lei Zhang
Microsoft Research Asia
4F Sigma, 49 Zhichun Road
Beijing, P.R.China
leizhang@microsoft.com

ABSTRACT

The method of finding high-quality answers has significant impact on user satisfaction in community question answering systems. However, due to the lexical gap between questions and answers as well as spam typically existing in user-generated content, filtering and ranking answers is very challenging. Previous solutions mainly focus on generating redundant features, or finding textual clues using machine learning techniques; none of them ever consider questions and their answers as relational data but instead model them as independent information. Moreover, they only consider the answers of the current question, and ignore any previous knowledge that would be helpful to bridge the lexical and semantic gap. We assume that answers are connected to their questions with various types of latent links, i.e. positive links indicating high-quality answers, negative links indicating incorrect answers or user-generated spam, and propose an analogical reasoning-based approach which measures the analogy between the new question-answer linkages and those of previous relevant knowledge which contains only positive links; the candidate answer which has the most analogous link is assumed to be the best answer. We conducted experiments based on 29.8 million Yahoo!Answer question-answer threads and showed the effectiveness of our approach.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*information filtering, search process*; G.3 [**Probability and Statistics**]: correlation and regression analysis; H.3.5 [**Information Storage and Retrieval**]: Online Information Services—*web-based services*

General Terms

Algorithms, Experimentation

*This work was performed in MSRA.

SIGIR'09, July 19–23, 2009, Boston, Massachusetts, USA.

Keywords

Community Question Answering, Analogical Reasoning, Probabilistic Relational Modeling, Ranking

1. INTRODUCTION

User-generated content (UGC) is one of the fastest-growing areas of the use of the Internet. Such content includes social question answering, social book marking, social networking, social video sharing, social photo sharing, etc. UGC web sites or portals providing these services not only connect users directly to their information needs, but also change everyday users from content consumers to content creators.

One type of UGC portals that has become very popular in recent years is the community question-answering (CQA) sites, which have attracted a large number of users both seeking and providing answers to a variety of questions on diverse subjects. For example, by December, 2007, one popular CQA site, Yahoo!Answers, had attracted 120 million users worldwide, and had 400 million answers to questions available [21]. A typical characteristic of such sites is that they allow anyone to post or answer any questions on any subject, which intuitively results in high variance in the quality of answers, e.g. UGC data typically has many content spam produced for fun or for profit. Thus, the ability, or inability, to obtain a high-quality answer has significant impact on user satisfaction [28].

Distinguishing high-quality answers from others in CQA sites is not a trivial task, e.g. a lexical gap typically exists between a question and its high-quality answers. The lexical gap in community questions and answers is caused by at least two factors: (1) textual mismatch between questions and answers; and (2) user-generated spam or flippant answers. In the first case, questions and answers are generally short, and the words that appear in a question are not necessarily repeated in its high-quality answers. Moreover, a word itself can be ambiguous or have multiple meanings, e.g., "apple" can either refer to "apple computer" or "apple the fruit". Meanwhile, the same concept can be described with different words, e.g. "car" and "automobile". In the second case, user-generated spam and flippant answers usually have a negative effect and greatly increase the number of answers, thereby make it difficult to identify the high-quality ones. Figure 1 gives several examples of the lexical gap between questions and answers.

To bridge the lexical gap for better answer ranking, various techniques have been proposed. Conventional tech-

niques for filtering answers primarily focus on generating complementary features provided by highly structured CQA sites [1, 4, 5, 14, 22], or finding textual clues using machine-learning techniques [2, 3, 19, 20], or identifying user authority via graph-based link analysis which assumes that authoritative users tend to generate high-quality answers [17].

There are two disadvantages of these work. Firstly, only the answers of the new question are taken into consideration. Intuitively it suffers greatly from the word ambiguity since questioners and answerers may use different words to describe the same objects (e.g. "car" and "automobile"). Secondly, questions and answers are assumed as independent information resources and their implicit correlations are ignored. We argue that questions and answers are relational. Recall the traditional QA approaches based on natural language processing (NLP) techniques [25] which attempted to discover natural language properties such as targeted answer format, targeted answer part-of-speech, etc., and used them as clues to figure out right answers. These are examples of implicit "semantic" clues, which is more valuable for right answer detection than the lexical clues suggested by terms.

We address these two problems in this study. In the offline stage, a large archive of questions and their answers are collected as a knowledge base. We then assume that there are various types of latent linkages between questions and answers, e.g. the high-quality answers are connected to their questions via semantic links, the spam or flippant answers are via noisy links, while low-quality or incorrect answers are through inferior links, etc. We denote the links associated with high-quality answers as "positive" links and the rest as "negative" ones and train a Bayesian logistic regression model for link prediction. By this means we are able to explicitly model the implicit q-a relationships. In the online stage, given a new question, we retrieve a set of relevant q-a pairs from the knowledge base with the hope that they would cover the vocabulary of the new question and its answers. These q-a pairs construct the prior knowledge on a similar topic to help bridge the lexical gap. Moreover, to discover the "semantic clues", instead of predicting directly based on the new question and its answers, we measure the analogy of a candidate linkage to the links embedded in the prior knowledge, and the most analogous link indicates the best answer. This is what we call "analogical reasoning(AR)".

Intuitively, taking an ideal case as an example, if the crawled knowledge base contains all questions in the world, to identify the best answer for a new question, we can just find the duplicate question in the knowledge base and use its best answer to evaluate a candidate answer's quality. However, since it is impractical to obtain all available questions as new questions are being submitted every day, we can instead search for a set of previously answered questions that best match the new question according to some specified metrics and discover analogous relationships between them. Intuitively, the retrieved similar questions and their best-answers are more likely to have words in common with the correct answers to the new question. Moreover, the retrieved set provides the knowledge of how the questions similar to the new question were answered, while the "way of answering" can be regarded as a kind of positive linkage between a question and its best answer.

We crawled 29.8 million Yahoo!Answers questions to evaluate our proposed approach. Each question has about 15 answers on average. We used 100,000 q-a pairs to train the link prediction model and tested the entire process with about 200,000 q-a pairs. We compared our method with two widely adopted information retrieval metrics and one state-of-the-art method, and achieved significant performance improvement in terms of average precision and mean reciprocal rank. These experimental results suggest that taking the structure of relational data into account is very helpful in the noisy environment of a CQA site. Moreover, the idea of leveraging previous knowledge to bridge the lexical gap and ranking a document with community intelligence is more effective than traditional approaches which only rely on individual intelligence.

Q: How do you pronounce the Congolese city Kinshasa?
A: "Kin" as in your family "sha" as in sharp "sa" as in sargent.

Q: What is the minimum positive real number in Matlab?
A: Your IQ.

Q: How will the sun enigma affect the earth?
A: Only God truly knows the mysteries of the sun, the universe and the heavens. They are His secrets!

Figure 1: Real examples from Yahoo!Answers, which suggest the lexical gap between questions and answers.

The paper is organized as follows. In Section 2, the most related work is discussed, which covers the state-of-the-art approaches on community-driven answer ranking. In Section 3, we detail the proposed approach. We evaluate its performance in Section 4, and discuss several factors which affect the ranking performance. We conclude our paper in Section 5 with discussions on future work.

2. RELATED WORK

2.1 Community Answer Quality Ranking

Different from traditional QA systems whose goal is to automatically generate an answer for the question of interest [25, 16], the goal of community answer ranking, however, is to identify from a closed list of candidate answers one or more that semantically answers the corresponding question. Since CQA sites have rich structures, e.g. question/answer associations, user-user interactions, user votes, etc., they offer more publicly available information than traditional QA domains.

Jeon et al. [14] extracted a number of non-textual features which cover the contextual information of questions and their answers, and proposed a language modeling-based retrieval model for processing these features in order to predict the quality of answers collected from a specific CQA service. Agichtein and his colleagues [1, 23] made great efforts on finding powerful features including structural, textual, and community features, and proposed a general classification framework for combining these heterogeneous features. Meanwhile, in addition to identifying answer quality, they also evaluated question quality and users' satisfaction. Blooma et al. [5] proposed more features, textual and non-textual, and used regression analyzers to generate predictive features for the best answer identification.

Some researchers resorted to machine learning techniques. Ko et al. [19] proposed a unified probabilistic answer rank-

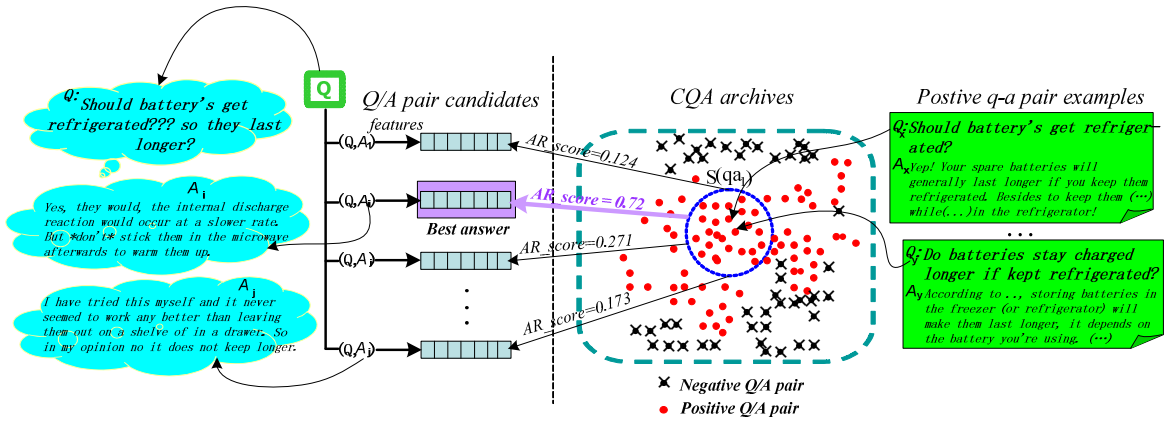


Figure 2: Sketch of the AR-based approach: 1)given a question Q (green block), some previous positive q-a pairs are retrieved (red dots in dotted blue circle); 2) each candidate q-a pair is scored according to how well it fits into the previous knowledge w.r.t. their linkages' properties. The magenta block highlights the highest-scored q-a pair candidate, which is assumed to contain the best answer.

ing model to simultaneously address the answer relevance and answer similarity problems. The model used logistic regression to estimate the probability that a candidate answer is correct given its relevance to the supporting evidence. A disadvantage of this model is that it considered each candidate answer separately. To solve this problem, the authors improved their solution and proposed a probabilistic graphical model to take into account the correlation of candidate answers [20]. It estimated the joint probability of the correctness of all answers, from which the probability of correctness of an individual answer can be inferred, and the performance was improved.

Bian et al. [3] presented the GBRank algorithm which utilizes users' interactions to retrieve relevant high-quality content in social media. It explored the mechanism to integrate relevance, user interaction, and community feedback information to find the right factual, well-formed content to answer a user's question. Then they improved the approach by explicitly considering the effect of malicious users' interactions, so that the ranking algorithm is more resilient to vote manipulation or "shilling" [2].

Instead of directly solving the answer ranking problem, some researchers proposed to find experts in communities with the assumption that authoritative users tend to produce high quality content. For example, Jurczyk et al. [17] adapted the HITS [18] algorithm to a CQA portal. They ran this algorithm on the user-answer graph extracted from online forums and showed promising results. Zhang et al. [30] further proposed ExpertiseRank to identify users with high expertise. They found that the expertise network is highly correlated to answer quality.

2.2 Probabilistic Relational Modeling

Probabilistic Relational Models [11] are Bayesian Networks which simultaneously consider the concepts of objects, their properties, and relations. Getoor et al. [10] incorporated models of link prediction in relational databases, and we adopt the same idea to learn the logistic regression model for link prediction.

3. THE APPROACH

3.1 Process Overview

Figure 2 shows the entire process of our approach. The red dots in the "CQA Archives" stand for "positive" q-a pairs whose answers are good, while the black crosses represent negative pairs whose answers are not good (not necessarily noisy). Each q-a pair is represented by a vector of textual and non-textual features as listed in Table 2.

In the offline stage, we learn a Bayesian logistic regression model based on the crawled QA archive, taking into account both positive and negative q-a pairs. The task of the model is to estimate how likely a q-a pair contains a good answer.

In the online stage, a supporting set (enclosed by the dotted blue circle in Figure 2) of positive q-a pairs is first retrieved from the CQA archives using only the new question Q (the green block) as a query. The supporting set, along with the learnt link prediction model, is used for scoring and ranking each new q-a pair, and the top-ranked q-a pair (the magenta block) contains the best answer. Some real q-a examples are given to better illustrate the idea.

3.2 Learning the Link Prediction Model

3.2.1 *Modeling latent linkage via logistic regression*

We train a Bayesian logistic regression (BLR) model with finite-dimensional parameters for latent linkage prediction and set multivariate Gaussian priors for the parameters. The advantage of introducing a prior is that it helps to integrate over function spaces.

Formally, let $X^{ij} = [\Phi_1(Q^i, A^j), \Phi_2(Q^i, A^j), \ldots, \Phi_K(Q^i, A^j)]$ be a K-dimensional feature vector of the pair of question Q^i and answer A^j, where Φ defines the mapping $\Phi : Q \times A \rightarrow R^K$. Let $C^{ij} \in \{0, 1\}$ be an indicator of linkage types, where $C^{ij} = 1$ indicates a positive linkage, i.e. A^j is a high-quality answer of Q^i, and $C^{ij} = 0$ otherwise. Let $\Theta = [\Theta_1, \Theta_2, \ldots, \Theta_K]$ be the parameter vector of the logistic regression model to be learnt, i.e.

$$P(C^{ij} = 1|X^{ij}, \Theta) = \frac{1}{1 + exp(\Theta^T X^{ij})} \quad (1)$$

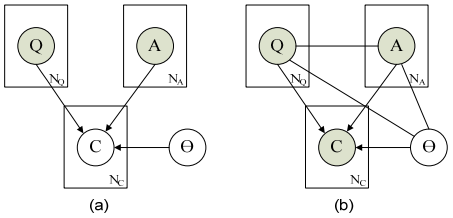


Figure 3: (a) The plate model of the Bayesian logistic regression model. (b) When $C = 1$, Q and A is positively linked which shares information embedded in Θ, represented by the undirected edges.

where $X^{ij} \in X_{train} = \{X^{ij}, 1 \leq i \leq D_q, 1 \leq j \leq D_a\}$ is a training q-a pair and D_q, D_a are the number of training questions and answers respectively. Generally we have $D_a \gg D_q$. Figure 3(a) shows the plate model. The C node is introduced to explicitly model the factors that link a question Q and an answer A. Since C represents a link while Q and A represent content, this model seamlessly reinforces the content and linkage in relational data. If $C = 1$ is observed, as shown in Figure 3(b), it means that there exists a positive link between the corresponding Q and A, and this link is analogous to links which construct Θ. We use undirected edges to represent these implications. To achieve a more discriminative formulation, we use both positive q-a pairs and negative q-a pairs to train this model. Meanwhile, since noisy answers generally occupy a larger population, to balance the number of positive and negative training data, we randomly sample a similar number of negative and positive points.

3.2.2 Learning the Gaussian prior $P(\Theta)$

The reason that we use a Gaussian prior is threefold: Firstly, instead of directly using the BLR model's outputs, we evaluate how probably the latent linkage embedded in a new q-a pair belongs to the subpopulation defined by the previous knowledge, with respect to the BLR prediction (we detail this approach in Section 3.4.2). Therefore Gaussian is a good prior candidate. Secondly, Gaussian distribution has a comparatively small parameter space (mean and variance), which makes the model easier for control. Thirdly, Gaussian distribution is conjugate to itself, i.e. the posterior is still a Gaussian, which results in a simple solution.

To ensure the preciseness of the prior, we adopt the approach suggested by Silva et al.[26]. Firstly the BLR is fit to the training data using Maximum Likelihood Estimation (MLE), which gives an initial $\hat{\Theta}$. Then the covariance matrix $\hat{\Sigma}$ is defined as a smoothed version of the MLE estimated covariance:

$$(\hat{\Sigma})^{-1} = \frac{c}{N} \cdot (\mathbf{X}^T \hat{\mathbf{W}} \mathbf{X}) \tag{2}$$

where c is a scalar; N is the size of the training data set; $\mathbf{X} = \{X^{ij}\}$ is the $N \times K$ feature matrix of the training q-a pairs, either positive or negative. $\hat{\mathbf{W}}$ is a diagonal matrix with $\hat{\mathbf{W}}_{\mathbf{ii}} = \hat{p}(i) \cdot (1 - \hat{p}(i))$, where $\hat{p}(i)$ is the predicted probability of a positive link for the ith row of $\mathbf{X}$.

The prior for Θ is then the Gaussian $\mathcal{N}(\hat{\Theta}, \hat{\Sigma})$.

3.3 Previous Knowledge Mining

To our knowledge, none of previous CQA answer ranking approaches ever leveraged a supporting set. In fact, such an idea is advantageous in at least two aspects: 1) bridging the lexical gap: it enlarges the vocabulary, which is more likely to cover the words not appearing in a question but in its correct answers or vice versa. In fact, some traditional QA approaches have discovered that the amount of implicit knowledge which associates an answer to a question can be quantitatively estimated by exploiting the redundancy in a large data set [6, 24]. 2) bridging the semantic gap: its positive linkages enable the analogy reasoning approach for new link prediction. Intuitively, it is more advantageous to rank a document based on community intelligence than simply on individual intelligence [7, 12].

We use information retrieval techniques to identify such a supporting set. However, community q-a pairs retrieval again is not a trivial task. Jeon and his colleagues [15] proposed a translation-based retrieval model using the textual similarity between answers to estimate the relevance of two questions. Xue et al [29], furthermore, combined the word-to-word translation probabilities of question-to-question retrieval and answer-to-answer retrieval. Lin et al. [22] and Bilotti et al. [4], on the other hand, adopted the traditional information retrieval method but utilized structural features to ensure retrieval precision.

We adopt Lin and Bilotti's way for q-a pair retrieval for simplicity. In particular, let Q_q be a new question and its answer list be $\{A_q^j, 1 \leq j \leq M\}$, the supporting q-a pairs $\mathbf{S} = \{(Q^1 : A^1), (Q^2 : A^2), \ldots, (Q^L : A^L)\}$ contain those whose questions' cosine similarities to the new question are above a threshold:

$$\mathbf{S} = \{(Q^i : A^i) | \cos(Q_q, Q^i) > \lambda, i \in 1, \ldots, D\} \tag{3}$$

where D is the size of our crawled Yahoo!Answer archive and λ is the threshold. Each question is represented in the bag-of-word model. The effect of λ is shown in Figure 5. As analyzed before, whether the retrieved questions are semantically similar to the new question is not critical. This is because the inference in our case is based on the structures of q-a pairs rather than on the contents. Moreover, according to the exhaustive experiments conducted by Dumais et al. [8], when the knowledge base is large enough, the accuracy of question answering can be promised with simple document ranking and n-gram extraction techniques. Note that $\mathbf{S}$ contains only positive q-a pairs. Therefore if the linkage of a candidate q-a pair is predicted as analogous to the linkages in the subpopulation $\mathbf{S}$, we can say that it is a positive linkage which indicates a high-quality answer.

3.4 Link Prediction

3.4.1 The idea of analogical reasoning

There is a large literature on analogical reasoning in artificial intelligence and psychology, which achieved great success in many domains including clustering, prediction, and dimensionality reduction. Interested readers can refer to French's survey [9]. However, few previous work ever applied analogical reasoning onto IR domain. Silva et al. [27] used it to model latent linkages among the relational objects contained in university webpages, such as student, course, department, staff, etc., and obtained promising result.

An analogy is defined as a measure of similarity between structures of related objects (q-a pairs in our case). The key aspect is that, typically, it is not so important how each individual object of a candidate pair is similar to individual

objects of the supporting set (i.e. the relevant previous q-a pairs in our case). Instead, implementations that rely on the similarity between the pairs of objects will be used to predict the existence of the relationships. In other words, similarities between two questions or two answers are not as important as the similarity between two q-a pairs.

Silva et al. [27] proposed a Bayesian analogical reasoning (BAR) framework, which uses a Bayesian model to evaluate if an object belongs to a concept or a cluster, given a few items from that cluster. We use an example to better illustrate the idea. Consider a social network of users, there are several diverse reasons that a user u links to a user v: they are friends, they joined the same communities, or they commented the same articles, etc. If we know the reasons, we can group the users into subpopulations. Unfortunately, these reasons are implicit; yet we are not completely in the dark: we are already given some subgroups of users which are representative of a few most important subpopulation, although it is unclear what reasons are underlying. The task now becomes to identify which other users belong to these subgroups. Instead of writing some simple query rules to explain the common properties of such subgroups, BAR solves a Bayesian inference problem to determine the probability that a particular user pair should be a member of a given subgroup, or say they are linked in an analogous way.

3.4.2 Answer ranking via analogical reasoning

The previous knowledge retrieved is just such a subpopulation to predict the membership of a new q-a pair. Since we keep only positive q-a pairs in this supporting set and high-quality answers generally answer a question semantically rather than lexically, it is more likely that a candidate q-a pair contains a high-quality answer when it is analogous to this supporting q-a set.

To measure such an analogy, we adopt the scoring function proposed by Silva et al. [27] which measures the marginal probability that a candidate q-a pair (Q_q, A_q^j) belongs to the subpopulation of previous knowledge $\mathbf{S}$:

$$\begin{aligned} score(Q_q, A_q^j) = &\log P(C_q^j = 1 | X_q^j, \mathbf{S}, \mathbf{C^S} = 1) \\ &- \log P(C^j = 1 | X_q^j)) \end{aligned} \tag{4}$$

where A_q^j is the j-th candidate answer of the new question. X_q^j represents the features of (Q_q, A_q^j). $\mathbf{C^S}$ is the vector of link indicators for $\mathbf{S}$, and $\mathbf{C^S} = 1$ indicates that all pairs in $\mathbf{S}$ is positively linked, i.e. $C^1 = 1, C^2 = 1, \ldots, C^L = 1$. The idea underlying is to measure to what extent (Q_q, A_q^j) would "fit into" $\mathbf{S}$, or to what extent $\mathbf{S}$ "explains" (Q_q, A_q^j). The more analogous it is to the supporting linkages, the more probability the candidate linkage is positive.

According to the Bayes Rule, the two probabilities in Eq.(4) can be solved by Eq.(5) and Eq.(6) respectively:

$$\begin{aligned} &P(C_q^j = 1 | X_q^j, \mathbf{S}, \mathbf{C^S} = 1) = \\ &\int P(C^j = 1 | X_q^j, \Theta) P(\Theta | \mathbf{S}, \mathbf{C^S} = 1) d\Theta \end{aligned} \tag{5}$$

$$P(C_q^j = 1 | X_q^j) = \int P(C^j = 1 | X_q^j, \Theta) P(\Theta) d\Theta \tag{6}$$

where Θ is the parameter set. $P(C^j = 1 | X_q^j, \Theta)$ is given by the BLR model defined in Eq.(1). The solution of these two equations is given in the appendix.

The entire process is shown in Table 1.

Table 1: Summary of our AR-based method

`I. Training Stage`:
Input: feature matrix $\mathbf{X_{train}}$ of all the training q-a pairs
Output: BLR model parameters Θ and its prior $P(\Theta)$.
Algorithm:

1. Train BLR model on $\mathbf{X_{train}}$ using Eq.(1).
2. Learn Prior $P(\Theta)$ using Eq.(2).

`II. Testing Stage`:
Input: a query QA thread: $\{Q_q : A_q^i\}_{i=1}^M$
Output: $score(Q_q, A_q^i), i = 1, \ldots, M$
Algorithm:

1. Generate the feature matrix $X_{test} = \{X_q^j\}$ with X_q^j the features of the j-th q-a pair $(Q_q : A_q^j)$;
2. Retrieve a set of positive q-a pairs $\mathbf{S}$ from the CQA archive using Eq.(3);
3. Do Bayesian inference to obtain $P(\Theta | S, C^S = 1)$;
4. For each (Q_q, A_q^j), estimate the probability of a positive linkage using Eq.(4);
5. Ranking the answers $\{A_q^j, j = 1, \ldots, M\}$ in descending order of the scores. The top-ranked answer is assumed as the best answer of Q_q.

4. EVALUATION

We crawled 29.8 million questions from the Yahoo!Answers web site; each question has 15.98 answers on average. These questions cover 1,550 leaf categories defined by expert and all of them have user-labeled "best answers", which is a good test bed to evaluate our approach. 100,000 randomly selected q-a pairs were used to train the link prediction model, and 16,000 q-a threads were used for testing — each contains 12.35 answers on average, which resulted in about 200,000 testing q-a pairs.

A typical characteristic of CQA sites is its rich structure which offers abundant meta-data. Previous work have shown the effectiveness of combining structural features with textual features [1, 14, 23]. We adopted a similar approach and defined about 20 features to represent a q-a pair, as listed in Table 2.

Two metrics were used for the evaluation. One is Average Precision@K: For a given query, it is the mean fraction of relevant answers ranked in the top K results; the higher the precision, the better the performance is. We use the "best answer" tagged by the Yahoo!Answers web site as the ground truth. Since average precision ignores the exact rank of a correct answer, we use the Mean Reciprocal Rank (MRR) metric for compensation. The MRR of an individual query is the reciprocal of the rank at which the first relevant answer is returned, or 0 if none of the top K results contain a relevant answer. The score for a set of queries is the mean of each individual query's reciprocal ranks:

$$MRR = \frac{1}{|Q_r|} \sum_{q \in Q_r} \frac{1}{r_q} \tag{7}$$

Table 2: The BAR Algorithm in CQA Formulation

Textual Features of Questions-Answers Pairs	
Q.(A.) TF	Question (Answer) term frequency, stopwords removed, stemmed
Novel word TF	Term frequency of non-dictionary word, e.g. "ms"
#Common-words	Number of common words in Q and A
Statistical Features of Questions-Answers Pairs	
Q.(A.) raw length	Number of words, stopwords not removed
Q/A raw length ratio	Question raw length / answer raw length
Q/A length ratio	Q/A length ratio, stopword removed
Q/A anti-stop ratio	#stopword-in-question/#stopword-in-answer
Common n-gram len.	Length of common n-grams in Q and A
#Answers	Number of answers to a question
Answer position	The position of an answer in the q-a thread
User interaction / Social Elements Features	
#Interesting mark	Number of votes mark a question as interesting
Good mark by users	Number of votes mark an answer as good
Bad mark by users	Number of votes mark an answer as bad
Rating by asker	The asker-assigned score to an answer
Thread life cycle	The time span between Q and its latest A

where Q_r is the set of test queries; r_q is the rank of the first relevant answer for the question q.

Three baseline methods were used: the Nearest Neighbor Measure (NN), the Cosine Distance Metric (COS), and the Bayesian Set Metric (BSets)[12]. The first two directly measure the similarity between a question and an answer, without using a supporting set; neither do they treat questions and answers as relational data but instead as independent information resources.

The BSets model uses the scoring function Eq.(8):

$$score(x) = \log p(x|\mathbf{S}) - \log p(x) \quad (8)$$

where x represents a new q-a pair and $\mathbf{S}$ is the supporting set. It measures how probably x belongs to $\mathbf{S}$. The difference of BSets to our AR-based method is that the former ignores the q-a correlations, but instead join the features of a question and an answer into a single row vector.

4.1 Performance

4.1.1 Average Precision@K

Figure 4 illustrates the performance of our method and the baselines with the Average Precion@K metric when $K = 1, 5, 10$. Since each question has less than 15 answers on average, the average precision at $K > 10$ is not evaluated.

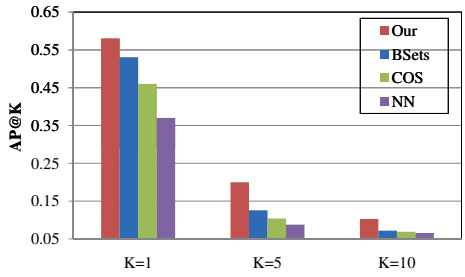


Figure 4: Average Precision@K. Our method significantly outperformed the baselines.

Table 3: MRR for Our Method and Baselines

Method	MRR	Method	MRR
NN	0.56	Cosine	0.59
BSets [12]	0.67	Our Method	**0.78**

The red bar shows the performance of our approach. The blue, yellow, and purple bars are of the BSets, COS and NN methods respectively. In all cases, our method significantly out-performed the baselines. The gap between our method and BSets shows the positive effect of modeling the relationships between questions and answers, while the gap between BSets and NN, COS shows the power of community intelligence.

The superiority of our model to BSets shows that modeling content as well as data structures improves the performance than modeling only content; this is because in CQA sites, questions are very diverse and the retrieved previous knowledge are not necessarily semantically similar to the query question (i.e. they are still noisy). Moreover, from the experimental result that NN method performed the worst, we can tell that the lexical gap between questions and answers, questions and questions, and q-a pairs cannot be ignored in the Yahoo!Answers archive.

4.1.2 Mean Reciprocal Rank (MRR)

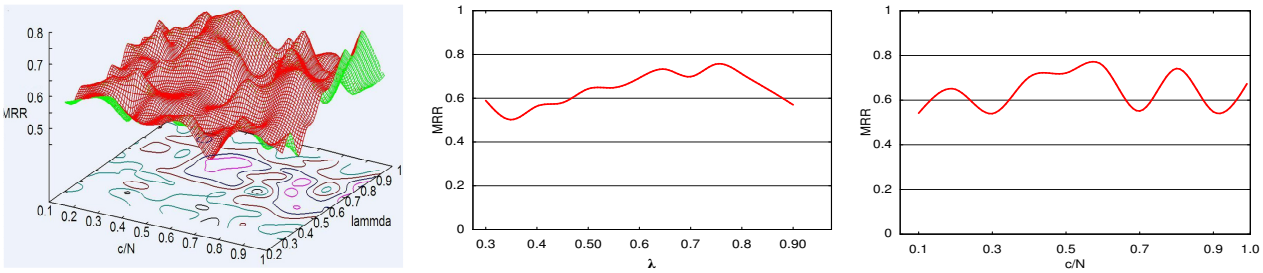
Table 3 gives the MRR performance of our method and the baselines. The trend coincides with the one that is suggested by the Average Precision@K measure. Again our method significantly out-performed the baseline methods, which means that generally the best answers rank higher than other answers in our method.

4.2 Effect of Parameters

Figure 5 evaluates how our method performs with the two parameters: $\frac{c}{N}$, the scalar in Eq.(2), and λ, the threshold in Eq.(3). MRR metric is used here.

Figure 5(a) shows the joint effect of these two parameters on the MRR performance. The best performance was achieved when $\lambda = 0.8, \frac{c}{N} = 0.6$. To better evaluate the individual effect of the parameters, we illustrate the curve of MRR vs. λ and MRR vs. $\frac{c}{N}$ in Figure 5(b) and Figure 5(c) respectively by fixing the other to its optimal value.

As described in Section 3.3, the threshold λ controls the relevance of the supporting q-a pair set. Intuitively, if λ is set too high, few q-a pairs will be retrieved. This causes too small a subpopulation of the supporting q-a pairs such that the linkage information becomes too sparse and thus is inadequate to predict the analogy of a candidate. On the



(a) The effect of $\frac{c}{N}$ and λ on the performance with MRR metric

(b) The trend of performance on MRR vs. λ. $\frac{c}{N} = 0.6$.

(c) The trend of performance on MRR vs. $\frac{c}{N}$. $\lambda = 0.8$.

Figure 5: The effect of the prior scalar $\frac{c}{N}$ in Eq.(2) and the similarity threshold λ in Eq.(3). The Mean Reciprocal Rank metric was used. The best performance was achieved at $\lambda = 0.8$, $\frac{c}{N} = 0.6$.

other hand, if λ is set too low, likely too many q-a pairs will be retrieved which will introduce too much noise into the subpopulation. Therefore the semantic meaning of the supporting links is obscured. Figure 5(b) exactly reflects such a common sense. The best performance was achieved when λ was about 0.8. After that the performance dropped quickly. However, before λ was increased to 0.8, the performance improved slowly and smoothly. This indicates that the noisy q-a pairs corresponding to diverse link semantics were removed gradually and the subpopulation was getting more and more focused, thus strengthened the power of analogical reasoning.

$\frac{c}{N}$ is used to scale the covariance matrix of the prior $P(\Theta)$. As shown in Figure 5(c), it is a sensitive parameter and is stable in a comparatively narrow range (i.e. about $0.4 \sim 0.6$). This, intuitively, coincides with the property of analogical reasoning approaches [9, 27, 26], where a data-dependent prior is quite important for the performance. In our approach, we used the same prior for the entire testing dataset which contains q-a pairs from diverse categories. A better performance can be foreseen if we learn different priors for different categories.

5. CONCLUSION

A typical characteristic of community question answering sites is the high variance of the quality of answers, while a mechanism to automatically detect a high-quality answer has a significant impact on users' satisfaction with such sites.

The typical challenge, however, lies in the lexical gap caused by textual mismatch between questions and answers as well as user-generated spam. Previous work mainly focuses on detecting powerful features, or finding textual clues using machine learning techniques, but none of them ever took into account previous knowledge and the relationships between questions and their answers.

Contrarily, we treated questions and their answers as relational data and proposed an analogical reasoning-based method to identify correct answers. We assume that there are various types of linkages which attach answers to their questions, and used a Bayesian logistic regression model for link prediction. Moreover, in order to bridge the lexical gap, we leverage a supporting q-a set whose questions are relevant to the new question and which contain only high-quality answers. This supporting set, together with the logistic regression model, is used to evaluate 1) how probably a new q-a pair has the same type of linkages as those in the supporting set, and 2) how strong it is. The candidate answer that has the strongest link to the new question is assumed as the best answer that semantically answers the question.

The evaluation on 29.8 million Yahoo!Answers q-a threads showed that our method significantly out-performed the baselines both in average precision and in mean reciprocal rank, which suggests that in the noisy environment of a CQA site, leveraging community intelligence as well as taking the structure of relational data into account are beneficial.

The current model only uses content to reinforce structures of relational data. In the future work, we would like to investigate how latent linkages reinforce content, and vice versa, with the hope of improved performance.

6. ACKNOWLEDGEMENT

We thank Chin-Yew Lin and Xinying Song's help on the q-a data.

7. REFERENCES

[1] E. Agichtein, C. Castillo, and etc. Finding high-quality content in social media. In *Proc. of WSDM*, 2008.

[2] J. Bian, Y. Liu, and etc. A few bad votes too many? towards robust ranking in social media. In *Proc. of AIRWeb*, 2008.

[3] J. Bian, Y. Liu, and etc. Finding the right facts in the crowd: Factoid question answering over social media. In *Proc. of WWW*, 2008.

[4] M. Bilotti, P. Ogilvie, and etc. Structured retrieval for question answering. In *Proc. of SIGIR*, 2007.

[5] M. Blooma, A. Chua, and D. Goh. A predictive framework for retrieving the best answer. In *Proc. of SAC*, 2008.

[6] E. Brill, J. Lin, M. Banko, and etc. Data-intensive question answeringa. In *TREC*, 2001.

[7] J. Chu-Carroll, K. Czuba, and etc. In question answering, two heads are better than one. In *Proc. of HLT/NAACL*, 2003.

[8] S. Dumais, M. Banko, and etc. Web question answering: Is more always better? In *Proc. of SIGIR*, 2002.

[9] R. French. The computational modeling of analogy-marking. *Trends in cognitive Sciences*, 6, 2002.

[10] L. Getoor, N. Friedman, and etc. Learning probabilistic models of link structure. *JMLR*, 3:679–707, 2002.

[11] L. Getoor, N. Friedman, and etc. Probabilistic relational models. *Introduction to Statistical Relational Learning*, 2007.

[12] Z. Ghahramani and K. Heller. Bayesian sets. In *Proc. of NIPS*, 2005.

[13] T. Jaakkola and M. Jordan. Bayesian parameter estimation via variational methods. *Statistics and Computing*, 10:25–37, 2000.

[14] J. Jeon, W. Croft, and et al. . a framework to predict the quality of answers with non-textual features. In *Proc. of SIGIR*, 2006.

[15] J. Jeon, W. Croft, and etc. Finding similar questions in large question and answer archives. In *Proc. of CIKM*, 2005.

[16] V. Jijkoun and M. Rijke. Retrieving answers from frequently asked questions pages on the web. In *Proc. of CIKM*, pages 76–83, 2005.

[17] P. Jurczyk and E. Agichtein. Discovering authorities in question answer communities by using link analysis. In *Proc. of CIKM*, 2007.

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

[19] J. Ko, L. Si, and E. Nyberg. A probabilistic framework for answer selection in question answering. In *Proc. of NAACL/HLT*, 2007.

[20] J. Ko, L. Si, and E. Nyberg. A probabilistic graphical model for joint answer ranking in question answering. In *Proc. of SIGIR*, 2007.

[21] J. Leibenluft. Librarianąŕs worst nightmare: Yahoo!answers, where 120 million users can be wrong. *Slate Magazine*, 2007.

[22] J. Lin and B. Katz. Question answering from the web using knowledge annotation and knowledge mining techniques. In *Proc. of CIKM*, 2003.

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

[24] B. Magnini and M. e. a. Negri. Is it the right answer? exploiting web redundancy for answer validation. In *Proc. of ACL*, pages 425–432, 2002.

[25] D. Molla and J. Vicedo. Question answering in restricted domains: An overview. In *Proc. of ACL*, 2007.

[26] R. Silva, E. Airoldi, and K. Heller. Small sets of interacting proteins suggest latent linkage mechanisms through analogical reasoning. In *Gatsby Technical Report, GCNU TR 2007-001*, 2007.

[27] R. Silva, K. Heller, and Z. Ghahramani. Analogical reasoning with relational bayesian sets. In *Proc. of AISTATS*, 2007.

[28] Q. Su, D. Pavlov, and etc. Internet-scale collection of human-reviewed data. In *Proc. of WWW*, pages 231–240, 2007.

[29] X. Xue, J. Jeon, and W. Croft. Retrieval models for question and answer archives. In *Proc. of SIGIR*, pages 475–482, 2008.

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

APPENDIX

Jaakkola et al. [13] suggested a variational approximation solution to the BLR model. Let $g(\xi)$ be the logistic function, $g(\xi) = (1 + e^{-\xi})^{-1}$, and consider the case for the single data point evaluation, Eq.(6), the method lower-bounds the integrand as follows:

$$\begin{aligned} P(C|X,\Theta) &= g(\Theta^T X) \\ &\geq g(\xi)\exp\{\frac{H_C - \xi}{2} - \lambda(\xi)(H_C^2 - \xi^2)\} \end{aligned} \tag{9}$$

where $H_C = (2C-1)\Theta^T X$ and $\lambda(\xi) = \frac{\tanh(\frac{\xi}{2})}{4\xi}$. $\tanh(\cdot)$ is the hyperbolic tangent function.

Thus $P(\Theta|X,C)$ can be approximated by normalizing

$$P(C|X,\Theta)P(\Theta) \geq Q(C|X,\Theta)P(\Theta)$$

where $Q(C|X,\Theta)$ is the right-hand side of Eq.(9).

Since this bound assumes a quadratic form as a function of Θ and our priors are Gaussian, the approximate posterior will be Gaussian, which we denote by $\mathcal{N}(\mu_{pos}, \Sigma_{pos})$. However, this bound can be loose unless a suitable value for the free parameter ξ is chosen. The key step in the approximation is then to optimize the bound with respect to ξ.

Let the Gaussian prior $P(\Theta)$ be denoted as $\mathcal{N}(\mu, \Sigma)$. The procedure reduces to an iterative optimization algorithm where for each step the following updates are made:

$$\begin{aligned} \Sigma_{pos}^{-1} &= \Sigma^{-1} + 2\lambda(\xi)XX^T \\ \mu_{pos} &= \Sigma_{pos}^{-1}[\Sigma^{-1}\mu + (C - \frac{1}{2})X] \\ \xi &= (X^T\Sigma_{pos}X + (X^T\mu_{pos})^2)^{\frac{1}{2}} \end{aligned}$$

To approximate Eq.(5), a sequential update is performed: starting from the prior $P(\Theta)$ for the first data point (X, C) in $(\mathbf{S}, \mathbf{C^S} = 1)$, the resulting posterior $\mathcal{N}(\mu_{pos}, \Sigma_{pos})$ is treated as the new prior for the next point. The ordering is chosen from an uniform distribution in our implementation.

Finally, given the optimized approximate posterior, the predictive integral Eq.(5) can be approximated as:

$$\begin{aligned} \log(Q(C^{ij}|X^{ij}, \mathbf{S}, \mathbf{C^S})) &= \log g(\xi_{ij}) - \frac{\xi_{ij}}{2} + \lambda(\xi_{ij})\xi_{ij}^2 \\ &-\frac{1}{2}\mu_{\mathbf{S}}^T\Sigma_{\mathbf{S}}^{-1}\mu_{\mathbf{S}}^T + \frac{1}{2}\mu_{ij}^T\Sigma_{ij}^{-1}\mu_{ij}^T + \frac{1}{2}\log\frac{|\Sigma_{ij}^{-1}|}{|\Sigma_{\mathbf{S}}^{-1}|} \end{aligned}$$

where parameters $(\mathbf{s}, \Sigma_{\mathbf{S}})$ are the ones in the approximate posterior $\Theta|(\mathbf{S}, \mathbf{C^S}) \sim \mathcal{N}(\mu_{\mathbf{S}}, \Sigma_{\mathbf{S}})$, and (μ_{ij}, Σ_{ij}) come from the approximate posterior $\Theta|(\mathbf{S}, \mathbf{C^S}, X^{ij}, C^{ij})$. Parameters (ξ_{ij}, ξ_S) come from the respective approximate posteriors.

And Eq.(6) can be approximated as:

$$\log Q(C^{ij}|X^{ij}) = \log g(\xi_{ij}) - \frac{xi_{ij}}{2} + \lambda(\xi_{ij})\xi_{ij}^2$$